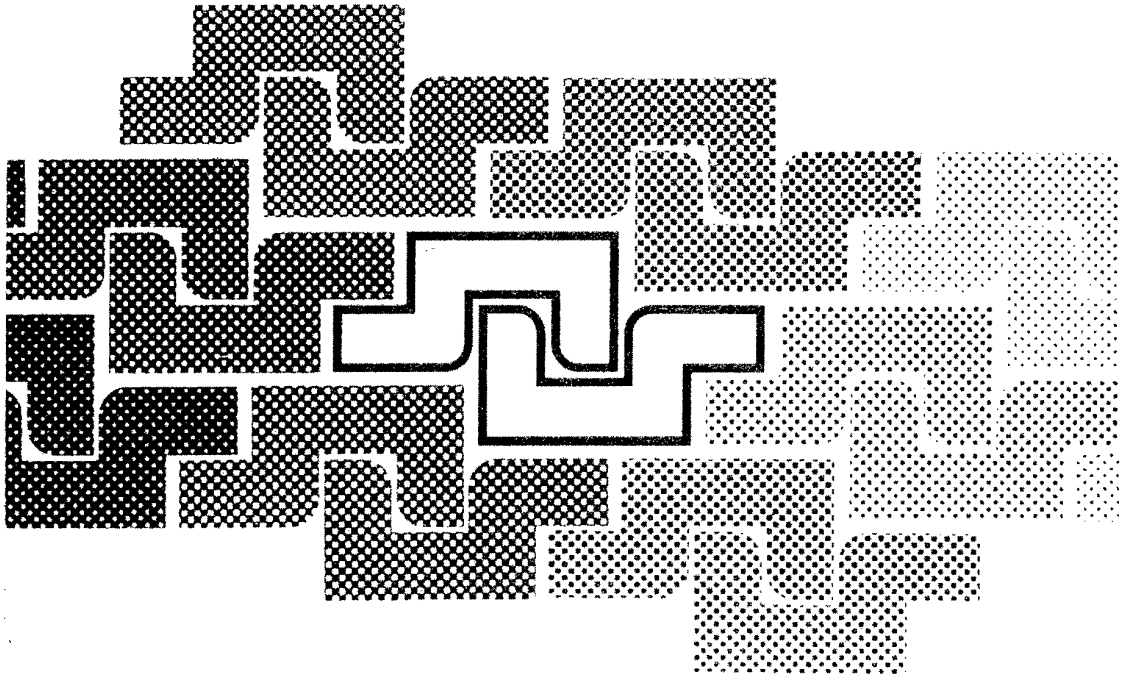


Virtual Nonlinear Multibody Systems

NATO Advanced Study Institute

Volume I

Edited by

Werner Schiehlen and Michael Valášek

Prague, June 23 – July 3, 2002

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	2002	Advanced Study Institute, 23 June – 3 July 2002

4. TITLE AND SUBTITLE

Virtual Nonlinear Multibody Systems, Volume 1

5. FUNDING NUMBERS

N62358-02-M-6031

6. AUTHOR(S)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Stuttgart, Germany

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES

USARDSG-UK, Fiscal Office, Edison House, 223 Old Marylebone Road, London NW1 5th, UK

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

R&D 9315-AN-02

11. SUPPLEMENTARY NOTES

NATO Advanced Study Institute on Virtual Nonlinear Multibody Systems, 23 Jun – 3 July 2002, Prague, Czech Republic, prepared under contract no. N62358-02-M-6031, Volume 1 is 252 pages and Volume 2 is 268 pages.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release.

12b. DISTRIBUTION CODE

A

ABSTRACT (Maximum 200 words)

Multibody system dynamics is based on classical mechanics and its engineering applications ranging from mechanisms, gyroscopes, satellites and robots to biomechanics and vehicle engineering. Multibody systems dynamics is characterized by algorithms or formalisms, respectively, ready for computer implementation. The simulation of multibody systems demands for adequate dynamic models and takes into account various phenomena. Classical dynamics does not regard all nonlinear effects that appear as a result of the action of multibody systems, as well as their mutual interaction. The virtual prototyping and dynamic modeling of such systems are, from an economical point of view, perspective fields of scientific investigations having in mind the huge expenses for their design and manufacturing. Complex multibody systems composed of rigid and flexible bodies performing spatial motion and various complex tasks are up-to-date objects of virtual prototyping. As a result simulation and animation featuring virtual reality are most important. Recent research fields in multibody dynamics include standardization of data, coupling with CAD systems, parameter identification, real-time animation, contact and impact problems, extension to electronic and mechatronic systems, optimal system design, strength analysis and interaction with fluids. Further, there is a strong interest on multibody systems in analytical and numerical mathematics resulting in reduction methods for the rigorous treatment of simple models and special integration codes for Ordinary Differential Equation (ODE) and Differential Algebraic Equation (DAE) representations supporting the numerical efficiency. New software engineering tools with modular approaches improve the efficiency still required for the more demanding needs in biomechanics, robotics and vehicle dynamics. The scientific research in multibody system dynamics is devoted to improvements in modeling considering nonholonomic constraints flexibility, friction, contact, impact and control. New methods evolved with respect to simulation by recursive formalism, to closed kinematic loops, reaction forces and torques, and pre- and post-processing by data models, CAD coupling, signal analysis, animation and strength evaluation. Multibody system dynamics is applied to a broad variety of engineering problems from aerospace to civil engineering, from vehicle design to micromechanical analysis, from robotics to biomechanics. In particular, multibody dynamics is considered as the basis of mechatronics, e.g. controlled mechanical systems. These challenging applications are subject to fundamental research topics which were presented at the NATO ASI on Virtual Nonlinear Multibody Systems.

14. SUBJECT TERMS

US Army Research, Czech Republic, Multibody system dynamics, Virtual prototyping, Dynamic modeling, Data models, CAD coupling, Parameter identification, Real-time animation evaluation, Strength evaluation, Software engineering, Signal analysis, Mechatronics

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited

Preprints of the
NATO Advanced Study Institute
on

Virtual Nonlinear Multibody Systems

Volume I

Edited by

Werner Schiehlen and Michael Valášek

Supported by the
European Research Office of the US Army

Czech Technical University in Prague
Prague, June 23 – July 3, 2002

Table of Contents

Simulation of Constrained Multibody Systems Based on Orthogonal Decomposition of Generalized Coordinates

Farhad Aghili and Jean-Claude Piedbœuf
({farhad.aghili,jean-claude.piedboeuf}@space.gc.ca)
Space Technologies, Canadian Space Agency, 6767 Route de l'Aéroport, St-Hubert, Québec, Canada, J3Y 8Y9

Abstract. This paper presents an efficient dynamic formulation for solving Differential Algebraic Equations (DAE) that is demanding in simulation of multibody systems containing constraint. The method is based on decomposition of the generalized coordinate into two orthogonal subspaces representing constraint coordinates and "self-motion" coordinate. The equation of motion of the self-motion coordinate is derived in an explicit form that is used to obtain the corresponding states as a result of numerical integration. The state associated with the constraint coordinates are obtained by solving algebraically the constraint equations.

1. Introduction

Dynamics of many mechanical systems is formulated as multibody system with closed-loop topology, e.g. manipulators interacting with environment, manipulators with closed-kinematic chain, parallel manipulators, vehicle and car suspension and steering system. Often one need to simulate such a complex dynamical system in real-time, thus speed and accuracy are important issues. Mathematically, simulation of constraint mechanical systems tantamount to solve a set of n differential equations coupled with a set of m algebraic equations, i.e. Differential Algebraic Equations (DAE). Equations describing a DAE system can be formally written as

$$\begin{cases} \mathbf{M}\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{A}^T\boldsymbol{\lambda} = \mathbf{f}. \\ \boldsymbol{\Phi}(\mathbf{q}, t) = 0 \end{cases} \tag{1}$$

Where $\mathbf{q} \in \mathbb{R}^n$ is the vector of generalized coordinate, $\mathbf{M} \in \mathbb{R}^{n\times n}$ is inertia matrix, $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$ contains Coriolis, centrifugal terms, $\boldsymbol{\lambda} \in \mathbb{R}^m$ is the Lagrangian multiplier corresponding to constraint force, $\mathbf{f} \in \mathbb{R}^n$ is vector of generalized force, and $\mathbf{A} \in \mathbb{R}^{m\times n}$ is the Jacobian of the constraint equation $\boldsymbol{\Phi} \in \mathbb{R}^m$ with respect to the generalized coordinates, i.e.

$$\mathbf{A} = \frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}}$$

It should be noted that, in general, the kinematic constraint $\Phi(\mathbf{q}, t) = 0$ is rhoeonomic reflecting the fact that the constraint condition is time-varying, i.e. some of the constraint equations depends on time. In solving equations (1), it is typically assumed that; (i) the mass matrix is invertible and (ii) the constraint equations are independent, i.e. the Jacobian matrix is not rank-deficient.

The survey of the existing techniques to solve DAE may be found in [4, 1, 5, 2, 3]. The classical method to deal with DAE is to express the constraint condition at acceleration level. This allows to replace the original DAE system with an ODE one by consolidating the two set of equations in (1), that is

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}^T \\ \mathbf{A} & 0 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} - \mathbf{h} \\ -\dot{\mathbf{A}}\dot{\mathbf{q}} - \dot{\mathbf{c}} + \mathbf{v} \end{bmatrix}, \tag{2}$$

where the augmented mass matrix is invertible if the Jacobian matrix is full rank, and hence $\ddot{\mathbf{q}}$ and $\boldsymbol{\lambda}$ can be obtained uniquely from the above matrix equation. Since maintaining the constraint at the acceleration level results in a marginally stable system, i.e. $\ddot{\Phi} = 0$, the Baumgarte stabilization [2] term $\mathbf{v} = -\mathbf{K}_v\dot{\Phi} - \mathbf{K}_p\Phi$ is used to ensure exponentially convergence of the constraint error to zero. The problem with this method though is that a very stiff system is required – by choosing high gains $\mathbf{K}_p$ and $\mathbf{K}_v$ – in order to keep small transient error in the constraint induced by numerical perturbation or by initial condition error. However choosing high gains creates virtually a very fast dynamics which slow down the simulation as maintaining stability of the integrator demands a small step size.

The other method is based on coordinate partitioning [8, 7] by making use of the fact that the n coordinates are not independent because of the m constraint equations. Consequently, at every instant the coordinate vector is partitioned as $\mathbf{q}^T = [\mathbf{q}^{dT}, \mathbf{q}^{iT}]$, where $\mathbf{q}^d \in \mathbb{R}^m$ and $\mathbf{q}^i \in \mathbb{R}^{n-m}$ are dependent and independent coordinates, respectively. The motion of the system described by the independent coordinates can be separated using an annihilator operator. Although this method may significantly reduce the number of equations, finding the annihilator operator is a very complex task [3].

In this work, we propose a new method for solving DAEs based on decomposing the generalized coordinates into two orthogonal subspaces using any generalized inverse of the constraint Jacobian. Unlike in the coordinate partitioning method, in this method the decomposed coordinates have no physical meaning and no order reduction is achieved. Nevertheless, automatic decomposition can be easily carried out based on any generalized inverse of the Jacobian. This paper is organized

as follows: section 2 describes the proposed decomposition algorithm. An explicit equation of motion describing the evolution of the self-motion coordinate is derived in section 3, while section 4 shows how the constraint states can be found algebraically from the kinematic equations.

2. Orthogonal Decomposition of Generalized Coordinate

By differentiating the constraint with respect to time, we have

$$\mathbf{A}\dot{\mathbf{q}} = -\mathbf{c} \quad \text{where} \quad \mathbf{c} = \frac{\partial \mathbf{\Phi}}{\partial t} \tag{3}$$

In (3) there are m set of nonlinear equations and n unknown, where $m < n$. That is there are fewer equations than unknowns. Therefore a family of solutions exist. The theory of linear system of equations establishes [6] that the general solution can be expressed by

$$\dot{\mathbf{q}} = \dot{\mathbf{q}}^c \oplus \dot{\mathbf{q}}^s$$

where $\dot{\mathbf{q}}^c$ is particular solution, associated with constraint equation (3), and $\dot{\mathbf{q}}^s$ is a homogeneous solution which belongs to the null space of the Jacobian matrix.

In this work we proposed a method for solving DAE by decomposing the velocity of the generalized coordinate into the two orthogonal subspaces; $\dot{\mathbf{q}}^s \in \mathcal{N}(\mathbf{A})$ (belong to the null space of the Jacobian), and $\dot{\mathbf{q}}^c \in \mathcal{S}(\mathbf{A})$ (belong to the support space of the Jacobian) – where $\mathcal{S}(\mathbf{A}) \equiv \mathcal{N}^{\perp}(\mathbf{A})$ and $\mathcal{N}(\mathbf{A}) \bigcup \mathcal{S}(\mathbf{A}) = \mathbb{R}^n$. In the sequel, the coordinates associated with $\dot{\mathbf{q}}^c$ and $\dot{\mathbf{q}}^s$ are called constraint coordinate and self-motion coordinate, respectively.

The projection of the generalized coordinates to the subspaces can be carried out by using projection operator

$$\dot{\mathbf{q}}^s = \mathbf{P}\dot{\mathbf{q}}, \tag{4}$$

$$\dot{\mathbf{q}}^c = (\mathbf{I} - \mathbf{P})\dot{\mathbf{q}}, \tag{5}$$

where

$$\mathbf{P} = \mathbf{I} - \mathbf{A}^{+}\mathbf{A} \tag{6}$$

can be calculated by using pseudo-inverse $\mathbf{A}^{+} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}$ of the Jacobian matrix [1].

[1] The pseudo-inverse and the projector operator can be also computed by the Singular Value Decomposition method. Suppose the Jacobian is written as $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, then the pseudo inverse can be found by $\mathbf{A}^{+} = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T$ and the projection is $\mathbf{P} = \mathbf{I} - \mathbf{V}\mathbf{V}^T$.

The velocity of the constraint coordinates $\dot{\mathbf{q}}^c$ can be obtained directly from the constraint equation (3), i.e.

$$\dot{\mathbf{q}}^c = -\mathbf{A}^+\mathbf{c}(t) \tag{7}$$

On the other hand, the homogenous solution, $\dot{\mathbf{q}}^s$ belonging to the null space, represents all self motion irrespectful of the constraint. However, the admissible motion is determined by the equation of motion of the self-motion coordinate that will be derived in the followings.

3. Equations of motion of the self-motion coordinate

To obtain equation of the Lagrangian multiplier, equations (1) can be solved first for an expression for acceleration

$$\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\mathbf{f} - \mathbf{h} - \mathbf{A}^T\boldsymbol{\lambda}) \tag{8}$$

which can be then substituted in the acceleration of the constraint equation,

$$\mathbf{A}\ddot{\mathbf{q}} + \dot{\mathbf{A}}\dot{\mathbf{q}} = -\dot{\mathbf{c}},$$

to obtain

$$\boldsymbol{\lambda} = (\mathbf{A}\mathbf{M}^{-1}\mathbf{A}^T)^{-1}\left[\mathbf{A}\mathbf{M}^{-1}(\mathbf{f} - \mathbf{h}) + \dot{\mathbf{A}}\dot{\mathbf{q}} + \dot{\mathbf{c}}\right] \tag{9}$$

Now by differentiating equation (4) with respect to time we have

$$\ddot{\mathbf{q}}^s = \mathbf{P}\ddot{\mathbf{q}} + \dot{\mathbf{P}}\dot{\mathbf{q}}. \tag{10}$$

In the following, we derive $\dot{\mathbf{P}}$ in an explicit form that is required to calculate the acceleration term of the above equation. By knowing that for any invertible matrix $\mathbf{B}$ we have

$$\frac{d}{dt}(\mathbf{B})^{-1} = -\mathbf{B}^{-1}\dot{\mathbf{B}}\mathbf{B}^{-1},$$

and that the matrix $\mathbf{A}\mathbf{A}^T$ is invertible (because the Jacobian is full rank), we can calculate the time derivative of the Jacobian pseudo-inverse as

$$\frac{d}{dt}[\mathbf{A}^+] = \mathbf{P}\dot{\mathbf{A}}^T(\mathbf{A}\mathbf{A}^T)^{-1} - \mathbf{A}^+\dot{\mathbf{A}}\mathbf{A}^+ \tag{11}$$

Substituting (11) in the time differentiation of equation (6) yields

$$\dot{\mathbf{P}} = -\mathbf{P}\dot{\mathbf{A}}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A} + \mathbf{A}^+\dot{\mathbf{A}}(\mathbf{A}^+\mathbf{A} - \mathbf{I})$$

By knowing that $\mathbf{P}^T = \mathbf{P}$ and by defining matrix $\mathbf{S} = \mathbf{Q}\mathbf{P}$, where

$$\mathbf{Q} = \mathbf{A}^+\dot{\mathbf{A}},$$

we have

$$\dot{\mathbf{P}} = -(\mathbf{S}^T + \mathbf{S}) \tag{12}$$

Finally, from equations (8), (10), and (12), we readily arrive at

$$\ddot{\mathbf{q}}^s = -\mathbf{P}\mathbf{M}^{-1}(\mathbf{h} - \mathbf{f} + \mathbf{A}^T\boldsymbol{\lambda}) - (\mathbf{S}^T + \mathbf{S})\dot{\mathbf{q}}. \tag{13}$$

This is the equation of acceleration of the self-motion coordinates in a closed form. Yet, in the following we rewrite equation (13) in a simpler form which has also some useful interpretations. One can show that $\mathbf{Q}^T\mathbf{P} \equiv 0$, hence from equation (7) we can say

$$\begin{aligned} \mathbf{Q}^T\dot{\mathbf{q}} &= \mathbf{Q}^T\dot{\mathbf{q}}^c \\ &= \mathbf{R}\mathbf{c}(t) \end{aligned} \tag{14}$$

where matrix $\mathbf{R}$ is defined such that

$$\mathbf{Q}^T = \mathbf{R}\mathbf{A}$$

Moreover, one can observe that $\mathbf{S}\dot{\mathbf{q}} = \mathbf{S}\dot{\mathbf{q}}^s$, hence equation (13) can be rewritten as

$$\ddot{\mathbf{q}}^s = -\mathbf{P}\left[\mathbf{M}^{-1}(\mathbf{h} - \mathbf{f} + \mathbf{A}^T\boldsymbol{\lambda}) - \mathbf{R}\mathbf{c}(t)\right] - \mathbf{Q}\dot{\mathbf{q}}^s \tag{15}$$

which expresses the equation of motion of the self-motion coordinates in a compact form.

It is interesting to point out that the first term and the second term of the acceleration (15) are in null space and support space of the Jacobian, respectively – note that $\mathbf{P}\mathbf{Q} \equiv 0$. Therefor, $\ddot{\mathbf{q}}^s \notin \mathcal{N}(\mathbf{A})$ unless the second term is identically zero. In that case the null set becomes time-invariant because the evolution of the velocity of the self-motion coordinate always takes place within that set. Since $\mathcal{N}(\mathbf{Q}) = \mathcal{N}(\dot{\mathbf{A}})$, we can say that the self-motion coordinate evolves in a time-invariant set if

$$\mathcal{N}(\mathbf{A}) \subseteq \mathcal{N}(\dot{\mathbf{A}}).$$

Then the second acceleration term, i.e. $\mathbf{Q}\dot{\mathbf{q}}^s$, is zero.

Equations (9) and (15) express the constraint force and acceleration of the self-motion coordinates in a closed form. Consequently, $\{\mathbf{q}^s, \dot{\mathbf{q}}^s\}$ can be obtained as a result of numerical integration. On the other hand the constraint coordinates can be algebraically derived from the constraint equation.

3.1. Scleronomic systems

Since $\mathcal{N}(\mathbf{R}) = \emptyset$, then

$$\mathbf{c} \equiv 0$$

is the only condition which vanishes the time-varying term $\mathbf{Rc}(t)$. Therefor, we can say that the time-varying term of the acceleration equation vanishes iff the constraint is time-invariant, i.e. Scleronomic constraint. Moreover, for a Scleronomic systems we have $\dot{\mathbf{q}}^c = \ddot{\mathbf{q}}^c = 0$, hence

$$\begin{cases} \dot{\mathbf{q}}^s = \dot{\mathbf{q}} \\ \ddot{\mathbf{q}}^s = \ddot{\mathbf{q}} \end{cases} \tag{16}$$

Therefore, the equation of motion of a Scleronomic system can be directly expressed in terms of the generalized coordinate, i.e.

$$\ddot{\mathbf{q}} = -\mathbf{P}\mathbf{M}^{-1}(\mathbf{h} - \mathbf{f} + \mathbf{A}^T\boldsymbol{\lambda}) - \mathbf{Q}\dot{\mathbf{q}} \tag{17}$$

4. The states of the constraint coordinate

Having obtained $\mathbf{q}^s$ and $\dot{\mathbf{q}}^s$ from integration of the acceleration, the kinematic constraint can be used to solve for $\mathbf{q}^c$ and $\dot{\mathbf{q}}^c$. Equation (7) gives the velocity of the constraint coordinates $\dot{\mathbf{q}}^c$. In essence, one should able to obtain the constraint coordinate from the constraint equation if the self-motion coordinate is known, i.e. $\mathbf{q}^c = \boldsymbol{\Omega}(\mathbf{q}^s, t)$. However, this explicit relationship may not exist. Hence the explicit nonlinear equation $\boldsymbol{\Phi}(\mathbf{q}^c, \mathbf{q}^s, t)$ should be solved numerically, e.g. the Newton-Raphson method, in terms of $\mathbf{q}^c$ where $\mathbf{q}^s$ is treated as a known parameter.

The Newton-Raphsom method solves a set of nonlinear equations iteratively based on linearized equations. Before we pay our attention to the linearized equation, we present some useful relationship. It can be inferred from (4) and (5) that $\frac{\partial \mathbf{q}^c}{\partial \mathbf{q}} = \mathbf{I} - \mathbf{P}$ and $\frac{\partial \mathbf{q}^s}{\partial \mathbf{q}} = \mathbf{P}$, hence by using the chain rule, we have

$$\mathbf{A} = \frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}^s}\mathbf{P} + \frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}^c}(\mathbf{I} - \mathbf{P})$$

By post multiplying the both sides of the above equation once with $\mathbf{A}^+$ and once with $\mathbf{P}$, we arrive at two equations:

$$\frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}^c}\mathbf{A}^+ = \mathbf{I} \tag{18}$$

$$\frac{\partial \boldsymbol{\Phi}}{\partial \mathbf{q}^s}\mathbf{P} = 0 \tag{19}$$

The constraint equation can be written in the so-called first-order differential form in multi variables, i.e.

$$\delta\Phi = \frac{\partial\Phi}{\partial\mathbf{q}^c}\delta\mathbf{q}^c + \frac{\partial\Phi}{\partial\mathbf{q}^s}\delta\mathbf{q}^s + \mathbf{c}\delta t$$

By virtue of (19) and $\delta\mathbf{q}^s \in \mathcal{N}(\mathbf{A})$, one can conclude that the second term in the RHS of the above equation is identically zero. This result was expected as the constraint condition is not to be affected by the self-motion coordinates. Now, by knowing the inverse of the Jacobian of the constraint with respect to $\mathbf{q}^c$ from (18), we can solve the linearized equation iteratively. The initial condition for the iteration loop is calculated by solving the time-varying part, i.e.

$$\{\mathbf{q}_0^c\}_{t+\delta t} = \{\mathbf{q}^c\}_t + \int_t^{t+\delta t} \mathbf{A}^+\mathbf{c}d\tau \tag{20}$$

Obviously the second term in the RHS of above equation is zero for Scleronomic constraints, that is the initial condition is equal to the value of the constraint coordinate at the previous step time. Thus the following loop

$$\mathbf{q}_{k+1}^c = \mathbf{q}_k^c - \mathbf{A}^+\Phi(\mathbf{q}_k^c, \mathbf{q}^s, t) \tag{21}$$

may be worked out iteratively until the error in the constraint falls into an acceptable tolerance, e.g. $\|\Phi\| \leq \epsilon$. It should be pointed out that the initial estimate given by (20) cannot be far from exact solution, because the drifting error within a single integration time step is quite small. Indeed, experiments have shown that even if the iteration loop (21) is called once every few time step, we still achieve a fast convergence.
Finally, the simulation of a constrained mechanical system based on the decomposition method can be proceed as the following steps:

1. compute the Lagrangian multiplier and the acceleration of self-motion coordinates from equations (9) and (15), assuming that the initial conditions are known.

2. obtain the self-motion states $\{\mathbf{q}^s, \dot{\mathbf{q}}^s\}$ as a result of numerical integration

3. use equations (7) and (21) to obtain the states associated with constraint coordinate $\{\mathbf{q}^c, \dot{\mathbf{q}}^c\}$. Having the vectors $\mathbf{q}$ and $\dot{\mathbf{q}}$ completely known, go to step(1).

Conclusion

In this paper we have presented an algorithm for solving DAEs. The method is based on orthogonal decomposition of generalized coordinates into two subspaces; the self-motion coordinates and constraint coordinates. Explicit equation of motion governing the dynamics of the self-motion coordinate has been derived that is used for simulation.

References

1. U. M. Ascher and L. R. Petzold. Computer methods for ordinary differential equations and differential-algebraic equations. SIAM, 1998.
2. J. Baumgarte. Stabilization of constraints and integrals of motion in dynamical systems. Computer Methods in Applied Mechanics and Engineering, 1:1–16, 1972.
3. A. Bichi, L. Pallottino, M. Bray, and P. Perdomi. Randomized parrallel simulation of constrained multibody systems for VR/Haptic applications. In IEEE Int. Conference on Robotics and Automation, pages 2319 – 2324, Seoul, Korea, 2001.
4. J. Garcia de Jalon and Eduardo Bayo. Kinematic and Dynamic Simulation of Multibody Systems. Springer-Verlag, 1989.
5. C. W. Gear. The simultaneous numerical solution of differential-algebraic equations. IEEE Trans. Circuit Theory, CT-18:89–95, 1971.
6. Y. Nakamura. Advanced Robotics: Redundancy and Optimization. Addison-Wesley Publishing Company, 1991.
7. P. E. Nikravesh and I. S. Chung. Application of euler parameters to the dynamic analysis of three dimensional constrained mechanical systems. ASME J. Mech. Design, 104:785–791, 1982.
8. R. A. Wehage and E. J. Haug. Generalized coordinate partitioning of dimension reduction in analysis of constrained dynamic systems. ASME J. Mech. Design, 104:247–255, 1982.

STATE-SPACE GENERALIZED PREDICITVE CONTROL FOR REDUNDANT PARALLEL ROBOTS

KVĚTOSLAV BELDA, JOSEF BÖHM, MICHAEL VALÁŠEK
Institute of Information Theory and Automation,
Academy of Sciences of the Czech Republic,
Department of Adaptive Systems.
Pod vodárenskou věží 4, 182 08 Praha 8 – Libeň, Czech Republic.

Keywords: Predictive control, Redundancy, Quadratic criterion, State formulation.

Abstract:

The paper deals with the design and properties of Generalized Predictive Control for path control of the redundant parallel robots. It summarizes classical and root minimization of the quadratic criterion and direct and two-step design of actuators respectively. As an example, the planar redundant parallel robot is used. Moreover, the paper presents several possibilities to use Predictive control for compliance of some additional requirements as smooth trends of actuators or fulfillment antibacklash condition.

1. Introduction

Topically, the next development in industrial area is constrained by deficit of powerful machines with proportional dynamics and stiffness. At the same time, the new control techniques, which would be able to achieve higher accuracy with keeping of dexterity of the robot constructions, are missing or, on the other hand, there is no interest for their real application during research and development of new machines.

One of the promising ways of solving mentioned problems is utilization of new robot type based on parallel construction [1]. However, this new concept of the constructions brings new questions, especially in control area, thus the parallel structures give the possibilities to significantly improve mechanical parameters of new machines (dexterity, dynamics, stiffness, kinematics' accuracy etc.).

The aim of this paper is investigation of one potential control approach - Generalized Predictive Control (GPC) [2, 3] as an example of high-level model-based control. This approach firstly offers to achieve higher accuracy (better compliance with technological requirements; i.e. for robots: better compliance of planned trajectory) and at the same time effective cooperation of all actuators – drives. Secondly it offers several possibilities to realize some additional requirements [4], e.g., requirement on smooth trends of actuators – drives or fulfillment antibacklash condition can be mentioned.

The paper initially focuses on model description of the parallel structure, then continues with introduction of predictive control technique and finally shows simulation examples and briefly discuses real - time application.

2. Description of the robot model

The robot (manipulator) is a multibody system, which can be described by Lagrange's equations, in redundant case, of mixed type. These equations lead to the DAE system (the Differential - Algebraic Equations) in the following form:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{s}} - \boldsymbol{\Phi}_s^T \boldsymbol{\lambda} &= \mathbf{g} + \mathbf{T}\mathbf{u} \\ \mathbf{f}(\mathbf{s}(t)) &= \mathbf{0} \end{aligned} \tag{1}$$

where $\mathbf{M}$ is a mass matrix, $\mathbf{s}$ is a vector of physical coordinates (their number is higher than number of degrees of freedom /DOF/), $\boldsymbol{\Phi}_s$ is an overall Jacobian of the system, $\boldsymbol{\lambda}$ are Lagrange's multipliers, $\mathbf{g}$ is a vector of right sides, matrix $\mathbf{T}$ transforms the inputs $\mathbf{u}$ (n torques) into n drives and $\mathbf{f}(\mathbf{s}(t)) = \mathbf{0}$ represents geometrical constrains.

The physical coordinates $\mathbf{s}$ consist of the independent coordinates $\mathbf{x}$ (Cartesian coordinates of the fix point of the cutting tool or gripper), drives'- actuators' coordinates $\mathbf{q}_1$ and other auxiliary geometrical coordinates $\mathbf{q}_2$.

Let us consider the possibility to transform the model (1) into independent coordinates $\mathbf{x}$ [5]. As follows, the DAE robot model is transformed to the ordinary differential model (ODE). It means that the Lagrange's multipliers disappear and design of the robot control becomes considerably simpler. Then the final model of the robot system is the following:

$$\mathbf{R}^T\mathbf{M}\mathbf{R}\ddot{\mathbf{x}} + \mathbf{R}^T\mathbf{M}\dot{\mathbf{R}}\dot{\mathbf{x}} = \mathbf{R}^T\mathbf{g} + \mathbf{R}^T\mathbf{T}\mathbf{u} \tag{2}$$

It is very important to note, firstly, that the Jacobian matrix $\mathbf{R}$ is the basis of the null space of the overall Jacobian $\boldsymbol{\Phi}_s$ and thus it satisfies the expression

$$\boldsymbol{\Phi}_s\,\mathbf{R} = \mathbf{R}^T\boldsymbol{\Phi}_s^T = \mathbf{0} \text{ and } \dot{\mathbf{s}} = \mathbf{R}\dot{\mathbf{x}} \rightarrow \ddot{\mathbf{s}} = \mathbf{R}\ddot{\mathbf{x}} + \dot{\mathbf{R}}\dot{\mathbf{x}} \tag{3}$$

and, secondly, the Jacobian $\mathbf{R}$ can be decomposed into submatrixes $\mathbf{R}_{q_1}$, $\mathbf{R}_{q_2}$ and $\mathbf{R}_x = \mathbf{I}_x$. Submatrix $\mathbf{R}_{q_1}$ ($= (\mathbf{R}^T\mathbf{T})^T$) defines important relation between $\dot{\mathbf{q}}_1$ and $\dot{\mathbf{x}}$ as

$$\dot{\mathbf{q}}_1 = \mathbf{R}_{q_1}\dot{\mathbf{x}} \quad \left(\equiv \frac{d\mathbf{q}_1}{dt} = \mathbf{R}_1 \cdot \frac{d\mathbf{x}}{dt}\right) \tag{4}$$

which will be useful in section dealing with design of control law in root form. $\mathbf{R}_1$ can be also obtained from geometrical relation $\mathbf{q}_1(\mathbf{x})$:

$$\mathbf{R}_1 = \left[\frac{\partial \mathbf{q}_1(\mathbf{x})}{\partial x_1}, \cdots, \frac{\partial \mathbf{q}_1(\mathbf{x})}{\partial x_n}\right]\Bigg|_{\substack{n = \text{number of independent} \\ \text{coordinates} = \\ = \text{degrees of freedom}}} \tag{5}$$

3. Classical design of control law

The principal task of control of the robots is accomplishment of their movement along a planned trajectory (technological requirements). In some cases, it is very sophisticated and difficult for general control approaches like classical PID structures. Therefore, the new control approaches, which are being developed, are directly adjusted for concrete system (machine, robot). High-level controls, which use knowledge of the mathematical model e.g. (1, 2), represent suitable approach, which takes into account dynamic trend of the controlled system. In this way, it can better comply with mentioned requirements from technology. On the basis of the dynamic model, equation (2), high level controls globally optimize whole control process and can predict future actions. One of them is Generalized Predictive Control (GPC).

The Predictive control [2,4] is a multi-step control based on local optimization of the quadratic criterion, where the linearized equation or state formula is used (i.e. only the nearest future control signal is evaluated). This approach admits combination of feedback~feedforward parts.

For design of predictive control law, the nonlinear model (2) must be linearized [5] and converted from continuous to discrete time. This described model transformation enables us to consider the classical discrete state formula in the following form:

$$\begin{aligned} \mathbf{X}(k+1) &= \mathbf{A}\,\mathbf{X}(k) + \mathbf{B}\,\mathbf{u}(k) \\ \mathbf{x}(k) &= \mathbf{C}\,\mathbf{X}(k) \end{aligned} \tag{6}$$

where $\mathbf{X}$ is composed as $\mathbf{X} = [\mathbf{x}, \dot{\mathbf{x}}]^T$ and $\mathbf{x}$ agrees with equation (2). Furthermore for law derivation, the expression of new unknown output values $\mathbf{x}$ from topical state $\mathbf{X}$ is needed. The following lines imply this expression.

$$\begin{aligned} \mathbf{x}(k) &= \mathbf{C}\mathbf{X}(k) \\ \hat{\mathbf{X}}(k+1) &= \mathbf{A}\ \mathbf{X}(k) + \quad \mathbf{B}\mathbf{u}(k) \\ \hat{\mathbf{x}}(k+1) &= \mathbf{C}\mathbf{A}\ \mathbf{X}(k) + \mathbf{C}\quad \mathbf{B}\mathbf{u}(k) \\ &\vdots \qquad\qquad \vdots \\ \hat{\mathbf{X}}(k+N) &= \mathbf{A}^N\mathbf{X}(k) + \mathbf{A}^{N-1}\mathbf{B}\mathbf{u}(k) + \cdots + \mathbf{B}\mathbf{u}(k+N-1) \\ \hat{\mathbf{x}}(k+N) &= \mathbf{C}\mathbf{A}^N\mathbf{X}(k) + \mathbf{C}\mathbf{A}^{N-1}\mathbf{B}\mathbf{u}(k) + \cdots + \mathbf{C}\mathbf{B}\mathbf{u}(k+N-1) \end{aligned}$$

then the prediction of $\mathbf{x}$ is the following

$$\hat{\mathbf{x}} = \mathbf{f} + \mathbf{G}\,\mathbf{u} \tag{7}$$

$$\mathbf{f} = \begin{bmatrix} \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^N \end{bmatrix} \mathbf{X}(k) \quad \text{and} \quad \mathbf{G} = \begin{bmatrix} \mathbf{C} & \mathbf{B} & \cdots & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{C}\mathbf{A}^{N-1}\mathbf{B} & & \cdots & \mathbf{C}\mathbf{B} \end{bmatrix} \tag{8}$$

Now we can optimize the quadratic criterion. The criterion is optimized in instant k, with using predictions of $\mathbf{x}$ ($\hat{\mathbf{x}} = [\hat{\mathbf{x}}_{k+1} \cdots \hat{\mathbf{x}}_{k+N}]^T$)

$$J_k = \mathcal{E}\left\{(\hat{\mathbf{x}} - \mathbf{w})^T(\hat{\mathbf{x}} - \mathbf{w}) + \mathbf{u}^T \boldsymbol{\lambda}\, \mathbf{u}\right\} = \\ = \mathcal{E}\left\{(\mathbf{G}\mathbf{u} + \mathbf{f} - \mathbf{w})^T(\mathbf{G}\mathbf{u} + \mathbf{f} - \mathbf{w}) + \mathbf{u}^T \boldsymbol{\lambda}\, \mathbf{u}\right\} \tag{9}$$

where $\mathcal{E}$ is an operator of mean value, N is a horizon of prediction, $\mathbf{x}$ is a vector of outputs, $\mathbf{w}$ are desired values, $\boldsymbol{\lambda}$ is a penalization of input and $\mathbf{u}$ is a vector of robot inputs. Considering the condition of optimization

$$J_k \stackrel{!}{=} \min \tag{10}$$

for criterion (9), the resultant control law is

$$\mathbf{u} = (\mathbf{G}^T\mathbf{G} + \boldsymbol{\lambda})^{-1}\mathbf{G}^T\,(\mathbf{w} - \mathbf{f}) \tag{11}$$

This control law (11) can be already used. It should be noted that only the first element $\mathbf{u}(k)$ from vector $\mathbf{u}$ is used. If penalization $\boldsymbol{\lambda}$ is greater than zero, then the matrix $\mathbf{G}^T\mathbf{G}$ is regular for all cases, adequately actuated even for redundant cases. Theoretical case of zero penalization $\boldsymbol{\lambda}$ with redundant robot can be solved by pseudoinversion [6].

4. Design of control law in root form

This chapter aims on derivation of control law for different configuration of elements in mathematical model (2), which needs matrices with smaller dimensions. Moreover, if the penalization is positive, the computation also holds the redundant properties (if exist). It can be also used for accomplishment of additional control requirements.

Furthermore, in this chapter, the advantages of the root optimization of quadratic criterion (9) are used, marked out by compact notation and good preparation for operations with huge matrices.

Let us proceed from nonlinear differential model (2) and from its simplified form:

$$\mathbf{R}^T\mathbf{M}\mathbf{R}\ddot{\mathbf{y}} + \mathbf{R}^T\mathbf{M}\dot{\mathbf{R}}\dot{\mathbf{y}} = \mathbf{R}^T\mathbf{g} + \mathbf{R}^T\mathbf{T}\mathbf{u} \\ \mathbf{R}^T\mathbf{M}\mathbf{R}\ddot{\mathbf{y}} + \mathbf{R}^T\mathbf{M}\dot{\mathbf{R}}\dot{\mathbf{y}} = \mathbf{R}^T\mathbf{g} + \mathbf{F}\mathbf{M} \tag{12}$$

where new vector **FM** represents new fictitious input to the system so called general forces.

In equation (12) we can apply the same procedures of linearization [5], discretization and use the same composition of prediction formula (chapter 3, $\hat{\mathbf{x}} = \mathbf{f} + \mathbf{G}\,\mathbf{u}$, (7)) for future output values.

The quadratic criterion (9) ($J_k = \mathcal{E}\left\{(\hat{\mathbf{x}}-\mathbf{w})^T(\hat{\mathbf{x}}-\mathbf{w})+\mathbf{u}^T\boldsymbol{\lambda}^T\boldsymbol{\lambda}\,\mathbf{u}\right\}$) can be rewritten in the root form as a product of matrices

$$J_k = \left[\,[\hat{\mathbf{x}}-\mathbf{w}]^T,\ \mathbf{u}^T\right]\begin{bmatrix}\mathbf{1} & \mathbf{0}\\ \mathbf{0} & \boldsymbol{\lambda}^T\end{bmatrix}\begin{bmatrix}\mathbf{1} & \mathbf{0}\\ \mathbf{0} & \boldsymbol{\lambda}\end{bmatrix}\begin{bmatrix}[\hat{\mathbf{x}}-\mathbf{w}]\\ \mathbf{u}\end{bmatrix} = \mathbf{J}^T\mathbf{J} \qquad (13)$$

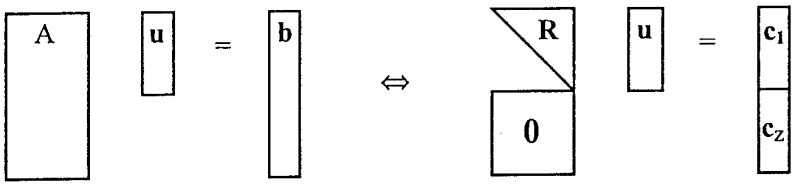
Now we can work only with root of the criterion

$$\mathbf{J} = \begin{bmatrix}\hat{\mathbf{x}}\\ \boldsymbol{\lambda}\,\mathbf{u}\end{bmatrix} - \begin{bmatrix}\mathbf{w}\\ \mathbf{0}\end{bmatrix} = \begin{bmatrix}\mathbf{G}\mathbf{u}+\mathbf{f}\\ \boldsymbol{\lambda}\,\mathbf{u}\end{bmatrix} - \begin{bmatrix}\mathbf{w}\\ \mathbf{0}\end{bmatrix} = \begin{bmatrix}\mathbf{G}\\ \boldsymbol{\lambda}\end{bmatrix}\mathbf{u} + \begin{bmatrix}\mathbf{f}\\ \mathbf{0}\end{bmatrix} - \begin{bmatrix}\mathbf{w}\\ \mathbf{0}\end{bmatrix}\left[=\mathbf{0}\right] \qquad (14)$$

$$\mathbf{A}\quad\mathbf{u}-\quad\mathbf{b}\quad=\mathbf{0} \qquad (15)$$

and consecutively we look for such action **u**, which minimizes root form (14, 15), i.e. we look for **u**, in order to minimize the norm |**J**|. If we annul the root of criterion (14), we will obtain system of equation (15) with more rows than columns (over-determined system).

For computation, the triangular-orthogonal decomposition [6] is used. It reduces excess rows of the matrix **A** $[(2{\cdot}N{\cdot}n)\times(N{\cdot}n)]$ and elements of vector **b** $[2{\cdot}N{\cdot}n]$ (n is a number of DOF) into upper triangle **R** and shorter vector $\mathbf{c}_1$ according to the following scheme:



(16)

Vector $\mathbf{c}_z$ is a loss vector. Its Euclidean norm $|\mathbf{c}_z|$ is equal to root of quadratic criterion; scalar $\sqrt{J}$ (i.e. $J = \mathbf{c}_z{}^T\mathbf{c}_z$).

For solution, we need only the upper part of the system of equations (16), which can be simply solved in view of the vector of actuators **u** by backward-run procedure.

Obtained actuators represent fictitious generalized force effects **u**, from which only the first subvector (for k instant) $\mathbf{u}(k)$ = **FM** is used. It must be recomputed, according to substitution in equations (12), to really used actions (drives):

$$\mathbf{R}^T\mathbf{T}\,\mathbf{u}_{(\text{drives})} = \mathbf{FM} \qquad (17)$$

with the same meaning of matrices **R** and **T** as in the system of differential equations (12). System (17) generally expresses deficient rank equation system (lower number of rows than columns i.e. than unknown real inputs - actions). There is again possibility to use pseudo-inverse of the matrix $\mathbf{R}^T\mathbf{T}$ there.

5. Examples and Conclusions

This section shows different actuators' trends for different control requirements, applied on planar redundant parallel robot (*Figure 1.*), for one selected trajectory.

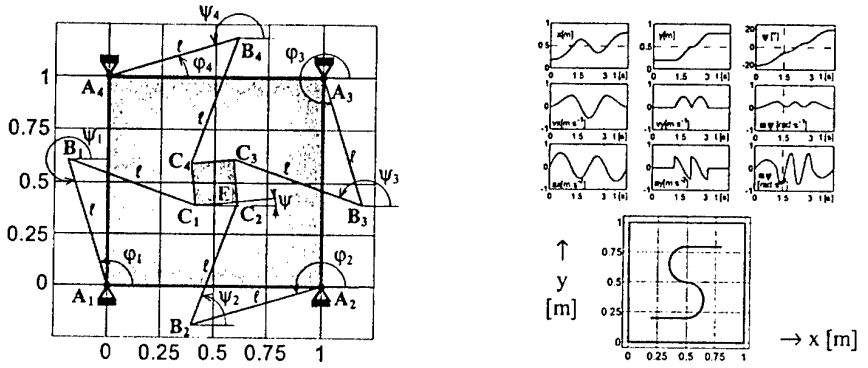


Figure 1. Scheme of the robot and example of the planned trajectory with its characteristics.

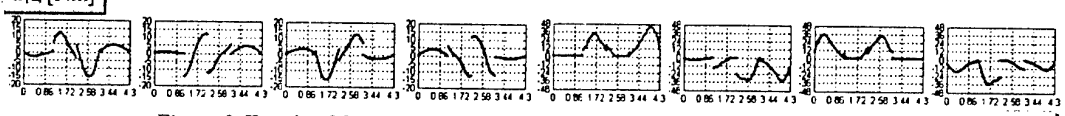


Figure 2. Trends of four actuators unlimited and limited by antibacklash condition (sampling Ts = 0.01s; max. error 1 μm; penalization $\lambda = 10^{-12}$; horizon $N = 10$).

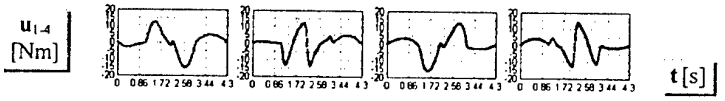


Figure 3. Smoothing of the actuators trends for trajectory from *Figure 1.* – variation of penalization (sampling Ts=0.01s; max. error 2.02 mm; penalization $\lambda = 5 \cdot 10^{-8}$; horizon $N = 10$).

The second, root approach is suitable for real application, because it represents less mathematical operations than classical approach. At present, it is tested on real laboratory model with the same structure as in *Figure 1.* As for result, both the approaches, classical and root control designs, are identical.

Acknowledgment

This research is supported by IGS ČVUT (CTU 0204512 - 1002045, 2002), "The study of the properties of independent (decentralized) and centralized control of redundant parallel robots".

6. References

1. Neugebauer, R. (ed.) (2002) *Development Methods and Application Experience of Parallel Kinematics*, Verlag Wissenschaftliche Scripten, Chemnitz, Germany.
2. Ordys, A. and Clarce, D. (1993) A state-space description for GPC controllers, *INT. J. Systems SCI.*, vol. 24, no. 9, 1727-1744.
3. Böhm, J., Belda, K. and Valášek, M. (2001) Study of control of planar redundant parallel robot. *Proceedings of the IASTED conference MIC 2001*, 694-699.
4. Böhm, J., Belda, K. and Valášek, M. (2001) The antibacklash task in the path control of redundant parallel robots. CD ROM Proc. of the 13th Int. Conf. Process Control 01'.
5. Valášek, M. and Steinbauer, P. (1999) Nonlinear control of multibody systems. *Euromech 404*, 437-444.
6. Lawson, Ch. and Hanson, R.J. (1974) *Solving least squares problems*, Prentice Hall.

A FIRST STEP TOWARDS FE MODELLING OF ERGONOMICS AND COMFORT

KAREL BENEŠ
MECAS ESI s. r. o.
Úslavská 10
32600 Plzeň
Czech Republic

1. INTRODUCTION

Scope. Ergonomic design and comfort of handling objects are often linked to the same activity of the human body, expressed in muscle action, depending on the arrangement and interaction of man and machine. Riding comfort depends on the mechanical aggression a given transport vehicle imparts on the individual, such as noise, vibration and harshness. Generally speaking, comfort may be linked to mechanical, acoustic, thermal, visual, psychological, etc. factors. Ergonomic and comfort design minimize the spent energy to perform an action (handling) and the effects of the inevitable aggression of various nature originating from an action (driving, etc.). In both cases fatigue of the muscles plays a major role when mechanical factors are involved. Psychological fatigue, however, may also play an important role in the actions of daily life and work. Here only those aspects of ergonomics and comfort are addressed that can be linked to mechanically induced muscle fatigue or to body reactions to external aggression, such as sustaining static loads and riding comfort.

Muscle action. The active forces of the muscles, for example, enable the human body to sustain a given position under static loads. These conscious actions also play a major role in the dynamic response of the human body subjected to dynamic loads and vibrations, such as from driving. In low energy car collisions, the muscle forces are known to have an influence on the injuries. A muscle can be kept at a given level of activation only for a certain period of time, where after the activation level involuntarily drops due to the physiological phenomenon of fatigue. Therefore, the activation levels of the muscular system can provide direct physical information towards the evaluation of comfort or ergonomics under the given circumstances. The "cost" of muscle action can be considered the product (or integral) of muscle force and time of action, which is to be minimized for comfort.

The H-ARB model. ESI has developed a human articulated rigid body (H-ARB™) model (Robby™), based on the skeletal geometry from Viewpoint DataLabs and corresponding to a 50-th percentile male human body. In a first part of a project, the complete muscular system for the arms, shoulders and neck has been implemented into

the skeleton. The muscles are represented by bars and are connected to the bones at their anatomical locations (points of origin and insertion). Their anatomical cross section, which determines the force they can develop at a given activation level, has been taken from different sources found in the literature and in anatomical atlases. The so generated "muscled" skeleton of the upper body can serve to evaluate the muscle forces for tasks involving the upper body.

PAM-Comfort™. ESI has developed a prototype software, where in the present first step of implementation the active force of each modeled muscle is determined for each loaded static position as the set of muscle forces that will sustain the imposed position in static equilibrium and that will also minimize the amount of spent energy. If dynamic inertia forces can be considered as equivalent static forces, solutions can also be found in such dynamic cases. Since the problem is statically over determined, direct solutions cannot be found. The solutions are therefore determined by an optimization algorithm, which calculates the active muscle (and external contact) forces acting on the articulated skeleton (design parameters) by minimizing the energy (objective function) under zero to full muscle activation levels (bounds) and for static equilibrium (constraints). Extra voluntary or involuntary muscle contractions beyond the levels necessary to equilibrate the imposed static loads can be taken into account by the elaborated software, when the level of extra contraction of the antagonist muscles is specified. Such bracing action may stiffen the skeletal kinematical chain, which may be beneficial in anticipation of shocks (car accidents) or imminent load peaks (weapon recoil), etc.

2. METHODOLOGY

Ergonomics, in its simplest expression, deals with the feasibility and comfort of humans performing tasks of quasi-static load carrying. A procedure to evaluate such simple scenarios is described. Possible extensions of the methodology can be to evaluate the optimal postures for the required task, or to evaluate optimal sequences of motions when performing a load carrying task.

Over-determined system. Since the number of kinematic degrees of freedom of the skeleton is far less than the number of muscle segments (bars) that can be activated to maintain a given static posture in equilibrium under a given static loading applied to the skeleton, the forces acting in each contributing muscle segment cannot be calculated from the mechanical conditions of equilibrium alone in a unique fashion. For this reason it is necessary to solve an over-determined system of equations by minimizing relevant objective functions that express the optimal involvement of each muscle segment that contributes to maintain the required posture under the applied static loads.

Hill's muscle model. The active and passive muscle actions are described by the well-known Hill muscle model, Figure 1. This model is valid for *quasi-static* extensions and contractions of skeletal muscles. In the case of suddenly applied loads to the skeleton, Hill's basic model is inadequate, because it does not provide for the correct dynamic stiffness of activated muscles. The Hill model was therefore augmented to include an *instantaneous dynamic* stiffness under high rates of change of muscle stretch. The

introduced stiffness is active for muscle stretch velocities large with respect to the fastest voluntary muscle contraction velocities. This stiffness was found roughly equal to the stiffness of the muscle tendon material spread over the length of the muscle and roughly proportional to the muscle activation level. The dynamic stiffness results from the instantaneous locking of the cross connected bridges between the myosine and actine fibers of each sarcomere.

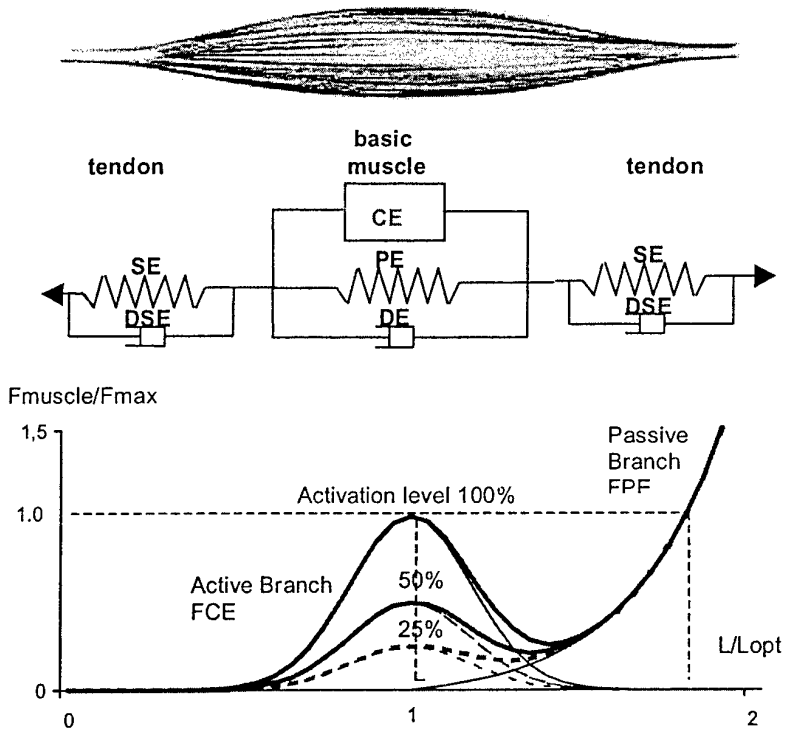


Figure 1. Hill's muscle model and it's maximal force/length dependency

Basic simulation methodology. Once the 'muscled' skeleton model has been established, positioned in the required static posture and loaded, assumptions are made on the intensities of the activated muscle-forces. These intensities are determined by the degree of voluntary muscle activation (0-100%) and are proportional to the cross sectional area of the considered muscle segment. For this purpose, likely agonists and antagonists ("prime movers") and synergizers and stabilizers ("assistants") are identified among the muscles, which participate in the investigated posture. The identified agonists are the main load carrying muscles, while the antagonists, if activated, directly counteract these muscles. The synergizers and the stabilizers play a secondary role. They hardly contribute to maintain a given posture under pure gravity loading, but assist the principal agonists under applied heavy loads.

A human subject can carry a given load in a given posture under more or less overall voluntary muscle contraction (0-100%). This can best be illustrated by the fact that an individual can willingly tense its muscles without carrying any load at all. In that case,

the agonist and antagonist muscles exactly balance their action on the skeleton, because otherwise the static position of zero motion will not be maintained. The load independent voluntary activation level of the antagonists can therefore be considered to represent the subject's level of voluntary muscle contraction in cases of applied static loads. This level of voluntary basic muscle activation can range from zero (most relaxed) to 100% (most contracted).

Optimization procedure. The *objective function*, assumed here to evaluate the likely distribution of the muscle forces to be activated in the principal agonists and secondary assistants has the form

$$f = \left(sum\,\gamma_i (\alpha_i - c)^2\right)^{\frac{1}{2}} \tag{1}$$

where the sum ranges over all participating muscle segments, i, activation level c is the given (average) voluntary level of muscle contraction before the load is applied (0-100%), α_i is the total activation level of the muscle segments that contribute to the task of carrying the load (0-100%) and γ_i is a switch which has the value "0" for the antagonists and the value "1" for the load carrying muscles (agonists, assistants). This function can be thought to express the *least possible overall level of muscle activation*, or "energy", to be expended for the task.

The *constraints* for the static optimization process are given by the fact that the *accelerations* of the links of the kinematic chain, constituted by the involved parts of the skeleton, must all be equal to *zero* in a position of static equilibrium. These accelerations can be calculated simply by performing an explicit analysis with the PAMSAFE solver code using the relevant muscled skeleton model with the applied loads. In fact one time step at time=0 is enough to determine whether or not the "structure" is in static equilibrium. At equilibrium, the internal muscle forces must balance the applied loads, and the accelerations, calculated by the solver at the centers of gravity of each rigid skeleton link, must be close to or equal to zero.

The *design parameters* of the optimization problem at hand are given by the *activation levels* of the participating *agonist* (and, perhaps more precisely, of the *assistant*) muscle segments. The activation level of a muscle cannot be less than zero and not greater than 100%. The outlined optimization procedure is applied to a simple one degree of freedom system.

3. ONE DEGREE OF FREEDOM SYSTEM

Test setup. Figure 1 shows an elementary one degree of freedom model and test setup of the upper and lower arm. The single kinematic degree of freedom consists in the rotation of the lower arm about the elbow joint with all other displacements and rotations fixed. The upper arm, the shoulder and the local wrist joints are considered fixed. From the 22 muscles of the upper and lower arm with a total of 28 segments, only the 2 segments of the biceps muscle, plus the brachialis and the brachioradialis muscles (4 segments) were retained as the agonists and the 3 segments of the triceps as the antagonist muscles. This reduced set constitutes a total of 7 muscle segments for one kinematic degree of freedom, i.e., the system to determine the muscle segment forces

from equilibrium is over-determined by a factor of 6.

The voluntary test subjects, were asked to pose their right elbow on a support (padded), to carry a load, P1, in the right hand and to voluntarily pretension the arm muscles to activation levels of zero, about 50% or 100%. At that moment, a second load, P2, initially suspended from the ceiling, was liberated by cutting its suspension string, whereupon the load P2 suddenly came into action at about the center of the lower arm. The subject's involuntary reactions due to this suddenly applied load were recorded on video. The reactions ranged from small angular responses (jerks) of the forearm for high voluntary muscle contraction to full, uncontrollable (unstable) extension of the forearm about the elbow joint for low or zero voluntary muscle contraction. The purpose of this test was to determine if the outlined procedure to "optimize" the muscle segment contributions in given static positions of equilibrium under applied skeleton loads can lead to plausible predictions of the forces, or activation levels, α_i of the agonists, when the antagonists undergo a constant pretension of c = 0%, 50% or 100% of their maximum activation. Since direct measurements of muscle forces were not possible (no electro-myographic apparatus was available), the activation levels could only be deduced indirectly by measuring the angular perturbations of the forearm about the elbow joint under the suddenly applied loads, P2. It was assumed that, if, for each applied load P1 and each level of muscular pre-stress, c, the simulation finds the same angular perturbations than were found in the tests, then the muscle force predictions can be considered accurate.

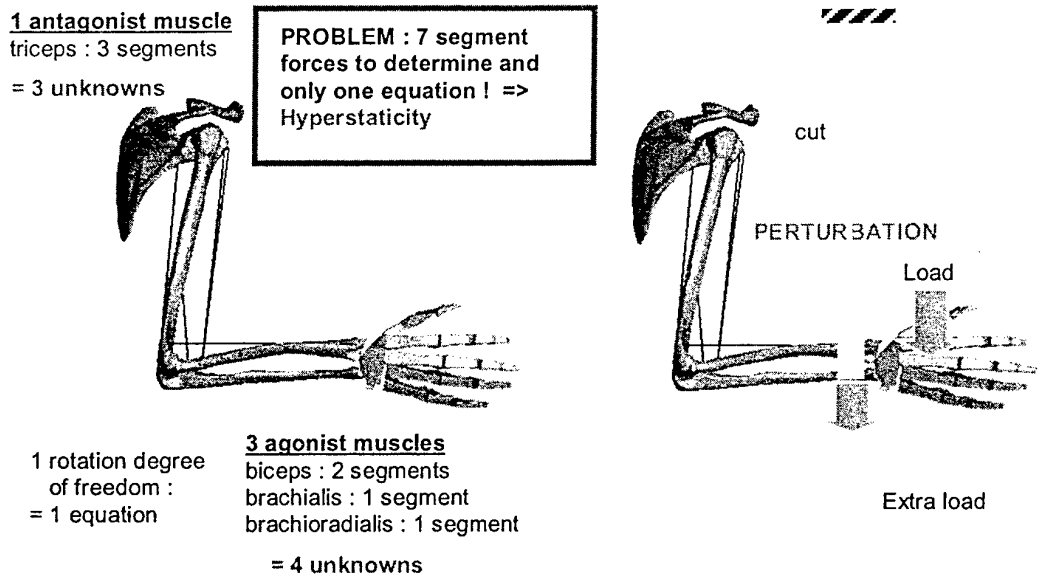


Figure 2. One-degree of freedom model of the elbow, and the usage of the perturbation technique

Test and simulation results. The preliminary results have shown that the test subject's responses to the suddenly applied extra loads could be predicted correctly, ranging from small extension angles to uncontrollable extension of the forearm. Since under the applied activation levels the simulations exhibited the same angular motions of the

forearm under the suddenly applied load of P2=4 kg force, it was concluded that the outlined procedure to determine the activation levels of the over-determined skeleton muscle system was realistic.

4. APPLICATIONS

Felin project. The outlined preliminary procedure has been applied in the Felin project of the French military to evaluate the performance of the musculo-skeletal system of humans in given postures under given static loads. Such problems arise when a mechanic is asked to hold in place a piece of heavy equipment in a hard to get at place (design problem), or when a military combatant is manipulating heavy equipment when loaded by unwieldy objects and gear. Based on the outlined procedure, criteria of "comfort" and "feasibility" may be deduced from the resulting necessary activation levels of the involved muscles.

Miscellaneous applications. The following pilot applications were investigated with the emerging PAM-Comfort ™ prototype numerical simulation tool: Gripping hand. Stowing of a bike. Sports. Different driver positions. Manipulating the hand brakes.

5. CONCLUSIONS

This document presents a short overview on the emerging ESI Group comfort and ergonomics models of the human body, that are developed to study the activation levels of the skeletal muscles, needed to sustain various load conditions. The shown examples indicate the wide spectrum of potential fields of application. The numerical methodology used to calculate the skeletal muscle forces proves to be remarkably efficient and leads in all studied cases to remarkably intuitive results. More validation studies must be performed, including electromyography measurements on volunteers. The models of the muscled skeleton must be completed for the still missing muscles, and scaling and morphing technologies must be used to produce models of different sizes. Extensions to dynamic forces and moving subjects are possible. The models are part of an emerging library of compute models in computational biomechanics ("BioLib": H-Model ™, Robby ™, etc.), which contains models of the human body that are conceived and validated mainly for studies of occupant safety of transport vehicles, comfort, ergonomics and biomedical applications. All models benefit from the synergy created from their different fields of application.

6. REFERENCES

[1] Schutz, T., "Etude d'un Modèle Ergonomique", Interim Report supervised by A. Trameçon, ESI SA, S. Pormenté and J.C. Craveur, ISMANS, August 1999.

SYMBOLIC AND SYSTEMATIC MULTIBODY MODELING FOR MECHATRONIC DESIGN

Dominique BERNIER, Antoine DEQUIDT, Etienne VALDES
Université de Valenciennes
Le Mont Houy F-59313 VALENCIENNES CEDEX 9
e-mail : dominique.bernier@univ-valenciennes.fr

1. Introduction

In mechatronic system design – Multi-Body System with drive system, sensor and control system, (e.g. robot, CNC machine tool, active suspension of vehicle) – the system model constitutes a virtual prototype for the behavior analysis, the validation and the optimization of parameters. But the use of model is not restricted with the final stage of design. During all the design process, different models (e.g. functional, kinematic, beam models) are used to build a technical solution gradually. This paper aims to describe the framework of the model generation of MBS for the process of design. It focus on the geometrical description of the open-loop structure, tree-structure and close-loop structure in a systematic way. An illustrative example is presented

2. Modeling system framework

The goal is to create a modeling tool allowing the description of a solution according to a design process stage (from the first stage to the last); the obtained model must be able easily to be modified and completed in a systematic way (without regenerating the whole of the model) according to the technical choices. Moreover, the model parameters will be used to the component sizing (e.g. cross section of a beam, an actuator) or the optimization. Lastly, the model can be used to describe various candidate solutions in order to evaluate, compare them and retain the best. The specificities of the modeling method described in this article are based on these design process characteristics.

Firstly, symbolic modeling system is used to fulfill the requirements stipulated before. This approach makes it possible to create a versatil tool of modeling for MBS applications [1]. It is convenient to create an open system of modeling of mechatronic system which makes it possible by example to complete a MBS model by models of particular physical phenomenon. That is appropriate well for the generation of control law, e.g. for robotics [2], and thereafter, for the simulation and feedback tuning with a numeric toolbox like MatLab [3]. From the point of view of the mechanical design (static, kinematic and dynamic criterium for technological choices, component sizing and parameter optimisation) the parameters of a symbolic model can be taken as variables of design.

Moreover, for a easy use by designer, the symbolic model must be obtained by using a systematic description method, which implies a nonredundant description and without ambiguity. In particular, the geometrical description of the kinematic structure is a crutial stage of modeling which will condition the generation of the dynamic model and the design. The Denavit & Hartenberg notation [4] is an efficient systematic description method with a minimum set of parameters but limited to the open-loop structure like serial robotic chains. This notation extended to tree-structure and close-loop structure by Khalil & Kleinfinger [5] is resumed and modified in the following part for the joints with 2, 3 or 4 degrees of freedom.

Lastly, the required goal is to create an object oriented modeling system. A model consists of a set of objects which can be modified or replaced by another independently to each other (without modifying the totality of the model). A Symbolic toolbox like Maple is an had hoc tool not only for the symbolic computation but also for the object oriented modeling [6]. The basic objects are the elementary

mechanical part models (building blocks for the description) : joint, body, actuator,... The main object of the MBS model is the kinematic pair, i.e. a joint plus a body, which is built with the basic objects. Its characteristics are fully or not fully defined in the kinematic pair object :

- the subscript of the pair,
- the subscript of the antecedent pair and of the next pairs,
- the kind of joints (revolute, spherical, Universal,... without or with strain and backlash),
- the geometry of the link between joints
- the body characteristics (for a rigid or flexible body)

This makes it possible to modify or replace easily an object within the kinematic structure during a design process. The kinematic topology can be established starting from the reading of the whole kinematic pair objects.

3. Systematic description of the kinematic structure

In robotics, there is a great diversity of methods making it possible to describe the position and the orientation of the elements of the kinematic chain: Denavit-Hartenberg method[4], formalism of Paul[7], modified Denavit-Hartenberg parameters[5], Seth-Uicker notation[8] ...
Each one of these formalisms was developed to describe kinematic chains with one degree of freedom joints (revolute or prismatic joint). Our concern was to take again a method not having ambiguity for the description of the robot-like chains and very usually used. According to our criteria, the best is the Denavit-Hartenberg method[4] modified by Khalil and Kleinfinger[5].
The method developed by Khalil et Kleinfinger[5] allows to write systematically and without ambiguity all the open-loop, tree and closed-loop structure robots with only one degree of freedom joints. To be able to describe the kinematic structures with a minimal parameter set, we have extended this method to spherical, universal, cylindrical, helical, prismatic-spherical and spherical-prismatic joints.

3.1. DESCRIPTION FOR AN OPEN-LOOP STRUCTURE

Unlike the other methods, we don't consider that a complex joint is a succession of simple joints with a null mass intermediate body. In comparison with Khalil and Kleinfinger method[5], we only add one intermediate frame and variable parameters are not more only supported by Z axis.
Our concern is also to be able to divide the transformation homogeneous matrices into two matrices: one to describe the geometry of the body and the second to describe the joint. This is why we have modified the initial method so that the subscript (i) of the parameters is relative to the joint (i) and to the body (i).
The joint (i) named L_i connects the body (i) to its antecedent (k) and the frame $R_i(O_i, X_i, Y_i, Z_i)$ is relative to the body (i). For all the joints with more than one degree of freedom, we have to add an additional frame named R_i^* (fig. 1).

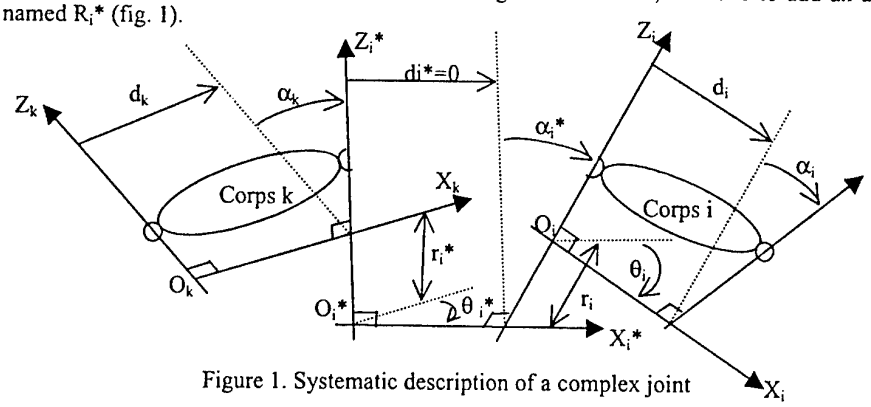


Figure 1. Systematic description of a complex joint

Z_i^* and Z_i axes will be choosen according to the type of joint (i) or by the user if the joint has no particular axis(for a shperical joint, Z_i^* axis can be parallel to Z_k and Z_i parallel to the Z axis of the succeding joint).
X axis is the common perpendicular to Z axis of the same frame and Z* axis of the next frame.
So, the frame R_i is defined with respect to the previous frame R_k by eight parameters :

- α_k angle between Z_k et Z_i^* about X_k,
- d_k distance between O_k et Z_i^*,

- θ_i^* angle between X_k et X_i^* about Z_i^*,
- r_i^* distance between X_k et O_i^*,
- α_i^* angle between Z_i^* et Z_i about X_i^*,
- d_i^* distance between O_i^* et Z_i , d_i^* is always equal to zero,
- θ_i angle between X_i^* et X_i about about Z_i,
- r_i distance between X_i^* et O_i.

α_k and d_k parameters are relative to the shape of the body (k) and the other parameters are relative to the joint (i). In the next table, we present particular values of parameters and generalized coordinates (relative coordinates) q for four types of joint.

Table 1 : parameters and generalized coordinates of some complex joints

	Spherical 3 d.o.f.	Universal joint 2 d.o.f.	Prismatic-spheric 4 d.o.f.	Spheric-prismatic 4 d.o.f.
α_k	0 or any			0 or any
d_k				
θ_i^*	$q_{i,\phi}$(precession euler angle)	$q_{i,1}$	$q_{i,\phi}$(precession euler angle)	$q_{i,\phi}$(precession euler angle)
r_i^*	0 or $f(\alpha_k)$		$q_{i,t}$	0 or $f(\alpha_k)$
α_i^*	$q_{i,\theta}$(nutation euler angle)	90° or other	$q_{i,\theta}$(nutation euler angle)	$q_{i,\theta}$(nutation euler angle)
d_i^*	0	0	0	0
θ_i	$q_{i,\psi}$(rotation euler angle)	$q_{i,2}$	$q_{i,\psi}$(rotation euler angle)	$q_{i,\psi}$(rotation euler angle)
r_i	0 or $f(\alpha_l)$		0 or $f(\alpha_i)$	q_4
α_i	0 or any		0 or any	
d_i				

3.2. DESCRIPTION FOR TREE-STRUCTURE AND FOR A CLOSE-LOOP STRUCTURE

In the case of a tree-structure chain(fig. 2), the body (i) can have several next bodies j and l. Then there is a common perpendicular by successor named $X_{i,0}$ and $X_{i,1}$, the second subscript indicating the name of the next body. So the parameters α_i and d_i become $\alpha_{i,0}$ and $d_{i,0}$ for primary branch. The user chooses one common perpendicular, let be (l) and sets up the parameters on this branch as for the simple chains. The other common perpendiculars are defined starting from the first by two parameters :

- $\varepsilon_{i,n}$ distance between $X_{i,0}$ and $X_{i,n}$ about Z_i,
- $\gamma_{i,n}$ angle between $X_{i,0}$ and $X_{i,n}$ about Z_i.

The other parameters ($\alpha_{i,n}$, $d_{i,n}$) on the secondary branches are defined like previously.

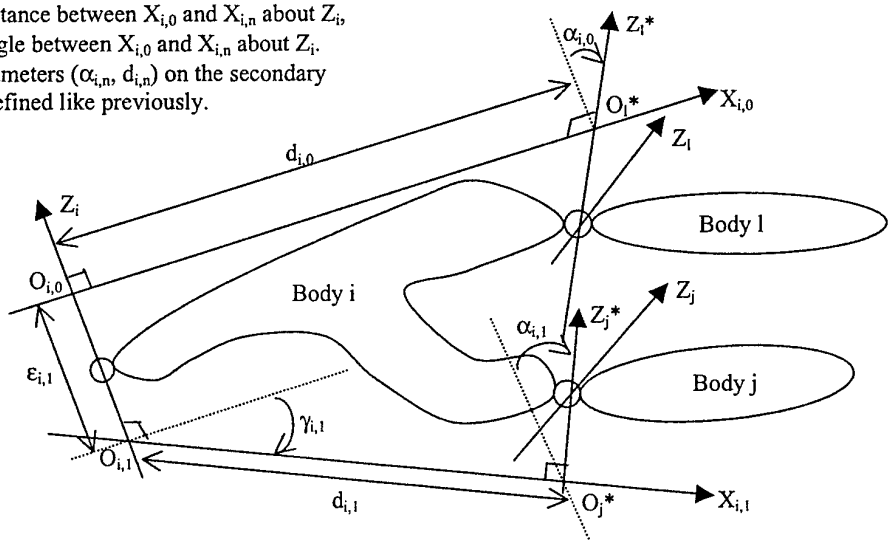


Figure 2 : Systematic description of a tree-structure

Parameters $\alpha_{i,0}$ and $d_{i,0}$ describe the shape of body (i) going to the joint (l) and parameters $\varepsilon_{i,l}$, $\gamma_{i,l}$, $\alpha_{i,l}$ and $d_{i,l}$ describe the shape of body (i) going to the joint (j).

In case of a close-loop structure, as Khalil and Kleinfinger [5], we open the loop by cutting a joint to define a tree equivalent structure and relations between the joint variables of the closed loop will be obtain by expressing the closure equation.

3.3. HOMOGENEOUS TRANSFORMATION MATRICES

For the main branch, the homogeneous transformation matrice ${}^{i}T_l$ between the frame R_i and R_l (fig. 2) can be written as the product of two matrices :

- one for the shape of body (i) going to the body (l), named $C_{i,0}$ function of $(\alpha_{i,0}, d_{i,0})$,
- one for the joint (l) named L_l function of $(\theta_l^*, r_l^*, \alpha_l^*, d_l^*, \theta_l, r_l)$,

$$ {}^{i}T_l = C_{i,0}L_l \tag{1}$$

For the secondary branches, the transformation matrice ${}^{i}T_j$ between the frame R_i and R_j (fig. 2) can be written as the product of two matrices :

- the shape of body (i) going to the body (j) matrice $C_{i,l}$ function of $(\varepsilon_{i,l}, \gamma_{i,l}, \alpha_{i,l}, d_{i,l})$.
- the joint matrice L_j, function of $(\theta_j^*, r_j^*, \alpha_j^*, d_j^*, \theta_j, r_j)$,

$$ {}^{i}T_j = C_{i,l}L_j \tag{2}$$

with:

$$C_{i,0} = \begin{bmatrix} 1 & 0 & 0 & d_i \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad C_{i,l} = \begin{bmatrix} \cos(\gamma_{i,l}) & -\sin(\gamma_{i,l})\cos(\alpha_{i,l}) & \sin(\gamma_{i,l})\sin(\alpha_{i,l}) & \cos(\gamma_{i,l})d_{i,l} \\ \sin(\gamma_{i,l}) & \cos(\gamma_{i,l})\cos(\alpha_{i,l}) & -\cos(\gamma_{i,l})\sin(\alpha_{i,l}) & \sin(\gamma_{i,l})d_{i,l} \\ 0 & \sin(\alpha_{i,l}) & \cos(\alpha_{i,l}) & \varepsilon_{i,l} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

and, by example for a spherical joint, with :

$$L_j = \begin{bmatrix} \cos(q_{j,\phi})\cos(q_{j,\psi}) - \sin(q_{j,\phi})\cos(q_{j,\theta})\sin(q_{j,\psi}) & -\cos(q_{j,\phi})\sin(q_{j,\psi}) - \sin(q_{j,\phi})\cos(q_{j,\theta})\cos(q_{j,\psi}) & \sin(q_{j,\phi})\sin(q_{j,\theta}) & 0 \\ \sin(q_{j,\phi})\cos(q_{j,\psi}) + \cos(q_{j,\phi})\cos(q_{j,\theta})\sin(q_{j,\psi}) & -\sin(q_{j,\phi})\sin(q_{j,\psi}) + \cos(q_{j,\phi})\cos(q_{j,\theta})\cos(q_{j,\psi}) & -\cos(q_{j,\phi})\sin(q_{j,\theta}) & 0 \\ \sin(q_{j,\theta})\sin(q_{j,\psi}) & \sin(q_{j,\theta})\cos(q_{j,\psi}) & \cos(q_{j,\theta}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

4. Illustrative example of the systematic description

The four-bar spatial mechanism (fig. 4), with body 0 to 3 and with Revolute, Universal, Spherical and Revolute joints, is a closed kinematic structure with one loop. The closed loop is cut on the joint between the bodies 0 and 3 in order to set up an equivalent open tree-structure with 2 branches (cf. fig. 3).

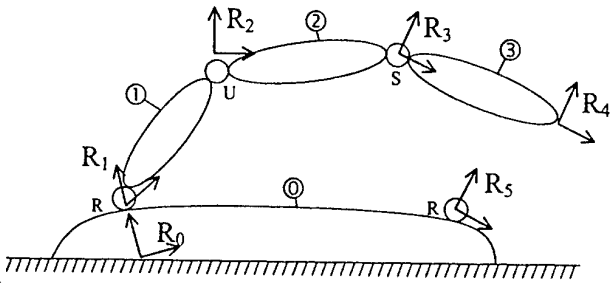


Fig. 3. open tree-structure corresponding to RUSR mechanism

The reference frames linked to each body (fig. 4) are posed in a systematic way in accordance with the rules described in part 3. In the same way, the systematic description of the geometry (fig. 5) leads to the table 2 of parameters and generalized coordinates.

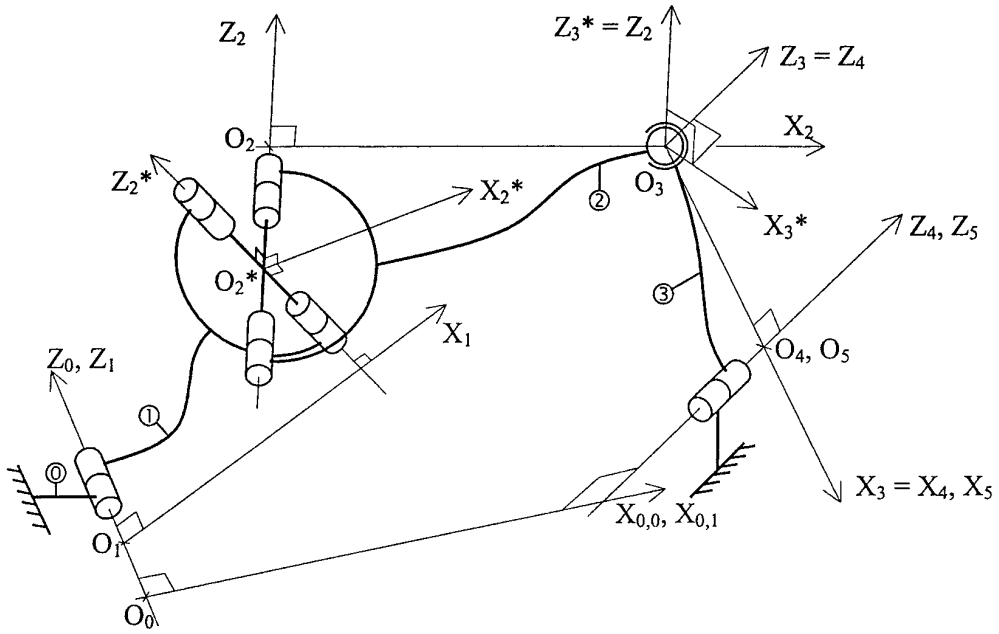


Fig. 4. Kinematic structure and reference frames of RUSR mechanism

Table 2 : Set of parameters and generalized coordinates of the open tree-structure

body subscript	Kind of joint	Joint			Body				succeding body subscript
		(θ^*	r^*	α^*)			α	d	
		θ	r		(ε_i	γ_i	α_i	d_i)	
0							0	0	1
					0	0	$\alpha_{0,1}$	$d_{0,1}$	5
1	R	q_1	r_1				α_1	d_1	2
2	U	$q_{2,1}$	r_2^*	$\pi/2$			0	d_2	3
		$q_{2,2}$	r_2						
3	S	$q_{3,\phi}$	0	$q_{3,\theta}$			0	d_3	
		$q_{3,\psi}$	0						
5	R	q_5	r_5						

The geometrical model of the mechanism is obtained starting from the homogeneous transformation matrices (1) and (2) associated with each branch of the tree structure :

$$^0T_4 = C_{0,0}\, L_1\, C_{1,0}\, L_2\, C_{2,0}\, L_3\, C_{3,0}\, L_4 \qquad (5)$$

$$^0T_5 = C_{0,1}\, L_5 \qquad (6)$$

with the matrice L_4 corresponding to fictitious joint in order to introduce the reference frame R_4 linked to the body 3.

$$L_4 = I \qquad (7)$$

The closure equation is obtained when one express that the reference frames R_4 and R_5 are identical :

$$^0T_4 = {}^0T_5 \qquad (8)$$

5. Conclusion

In this paper, we have presented a method to describe kinematical structures with more than one degree of freedom joints. The advantages of this method is to be systematic and non-ambiguous, and it allows to obtain a minimum parameter set in order to be used in the model generation of MBS framework, notably to determinate its dynamical model [9] in context of mechatronic design.

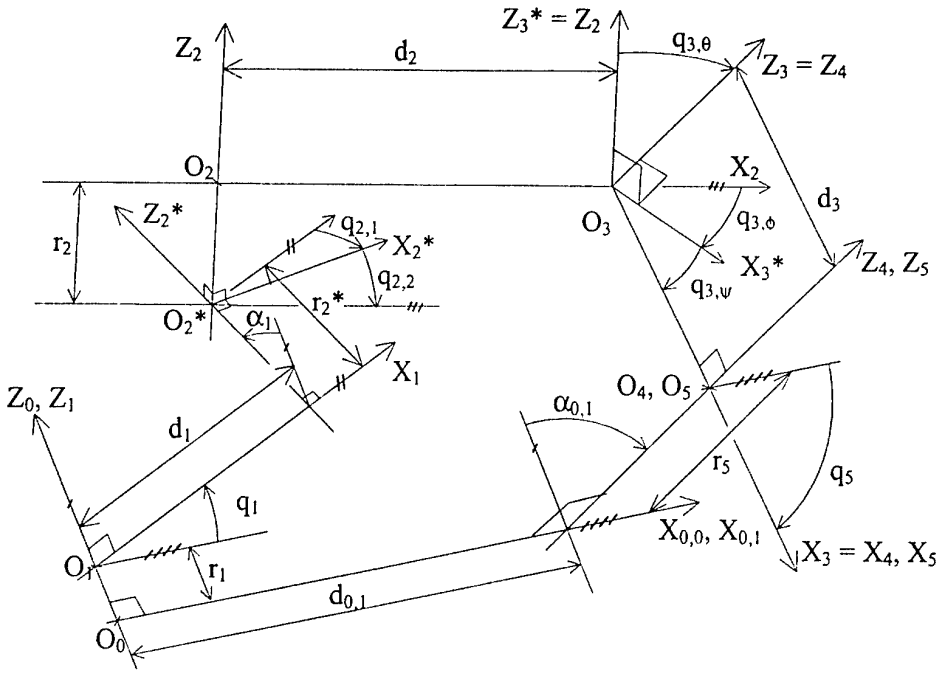


Fig. 5. Reference frame and geometric parameters of the RUSR mechanism

6. References

1. Chenut, X., Fisette, P., Samin, J.C. (1997) The benefits of symbolic generation for studying mechanical and mechatronic systems, *Proc of the NATO ASI on Computational Methods for Mechanisms, Varna, Bulgaria*, **2**, pp. 61-70.
2. Khalil, W., Creusot, D. (1997) SYMORO+ : a system for symbolic modelling of robots, *Robotica* **15**, pp. 153-161.
3. Gautier, M., Pham, M. T., Khalil, W., Lemoine, Ph., Poignet, Ph. (2000) SICOMAT : A system for SImulation and COntrol analysis of MAchine Tools, *II Int. Seminar on Improving Machine Tool Performance, La Baule, France,* Paper n° A6 (CD Rom).
4. Denavit, J., Hartenberg, R.S. (1955) A kinematic notation for lower pair mechanism based on matrices, *Trans of ASME J. of Applied Mechanics*, **22**, pp. 215-221.
5. Khalil, W., Kleinfinger, J.-F. (1986) A new geometric notation for open and closed-loop robots, *Proc IEEE Int. Conf. on Robotics and Automation, San Francisco, April,* pp. 1174-1180.
6. Mourrain, B., Papagay, Y. (1997) *Ecole de printemps de calcul formel, 14-18 Avril, Sophia Antipolis,* INRIA.
7. Paul, R. P. (1981) *Robot Manipulators : Mathematics, Programming, and Control*, The MIT Press.
8. Sheth, P.N., Uicker, J.J. (1971) A generalized symbolic notation for mechanism, *Trans of ASME J. of Engineering for Industry*, **93**, pp. 102-112.
9. Kleinfinger, J.-F., Khalil, W., (1986) Dynamic modelling of closed-chain robots, *Proc 16th Int. Symp. On Industrial Robots, Bruxelles, sept.-oct.*, pp. 401-412.

DYNAMIC SENSIBILITY OF MECHATRONIC SYSTEMS WITH REDUNDANCY

G.V. BOIADJIEV, D.B. VASSILEVA
Institute of Mechanics – Bulgarian Academy of Sciences
Acad. G. Bonchev Str., bl. 4, 1113 Sofia, Bulgaria

1. Introduction

Mechatronic systems as general are hybrid ones, which consist of subsystems having different physical nature. Mainly they include mechanical, electrical and electronic parts, sensor and computer devices, control system etc. Robot manipulators are typical examples for a mechatronic system. Having a complex structure themselves they could be applied for various tasks solution, but some other possibilities are not paid attention enough concerning the inner resources, the additional storage of the manipulative structures already designed. The last traces the way of new constructions and the considerations resulting in very interesting characteristics and parameters improving theoretical understanding as well as helpful for optimizing the practical applications as a whole. Such characteristics related with the mechatronic system state is the sensibility, which is a system quality characteristic having several quantity parameters. The presence of redundancy reflects the larger possibilities for optimizing the sensibility parameters.

Purpose. The purpose of the work is sensibility analysis of mechatronic system's mechanical subsystem - robot manipulators as well as redundancy influence on kinematic and dynamic sensibility parameters for accuracy and energy optimisation.

2. Kinematic sensibility.

The kinematic sensibility is a system quality characteristic having as quantity parameters corresponding sensibility coefficients and directions. It can be described mathematically by transformation τ mapping the configuration robot space $Q \in R^n$ to its working one R^3. The transformation τ, is a homomorphism, consisting of two different ones τ_p and τ_r. They map the neighbourhood ΔQ round the point (configuration) $q \in Q$ into the sensibility position and orientation ellipsoids. The center of each of them coincides with the point q and their semi-axes, following the sensibility directions, are equal to the sensibility coefficients by absolute values. The coefficients and directions are obtained as solutions of general task of eigenvectors for both homomorphisms. Obviously, the rank of τ_p and τ_r does not exceed the dimension of R^3. The presence of redundancy reflects the dimension of Q, i.e. it becomes bigger.

2.1. POSITIONING

Tree-like manipulative structures are considered with n degrees of freedom where contiguous bodies are connected by translation and rotational joints. The joint parameters q_i ($i = 1,....,n$) are chosen as generalized coordinates. The vectors $q = (q_1,...,q_n)^T$ belong to the

configuration space. An arbitrary point H is fixed in the last structure's body. Two coordinate systems are fixed in the support and in the last structure's body. Usually in practice the needed state realizes some deviations δR and $\delta\theta$ having probability behaviour. This is due to various reasons - errors in geometry, errors in calculations, compliance, sensing, calculations etc.

In the case of position the deviations are described by the following expression

$$\delta R = A(q)\delta q\,, \qquad (1)$$

Let's consider an ε - neighbourhood round a configuration q and assume the vectors δq belong there. It is proved [5], [1] the transformation (1) maps n-dimensoinal ball ε in k-dimensional sensibility ellipsoid $E_p \in R^3$, where k = rankA. It is also shown [2], [3] the ellipsoid's semi-axes' lengths are upper borders of $/\delta R/$ on k orthogonal directions and they are obtained as eigenvalues of general task of eigenvectors: $(B_p - \lambda C)X = 0$. For every state q the matrix A(q) from (1) defines a homomorphism τ_p between configuration space Q and working zone Z. The image of τ_p is the sensibility ellipsoid for positioning [7] and the kernel is its orthogonal completing.

2.2. ORIENTATION

In the case of orientation the deviations are described by the following expression, which is equivalent to (1):

$$\delta\theta = L(q)\delta q \qquad (2)$$

For every state $q \in Q$, the matrix L(q) from (2) defines a homomorphism τ_r between the configuration space Q and the working zone Z.

3. Dynamic modelling

In practical application the real motion is going under different force constraints, especially the contact tasks. So, the sensibility analysis has to be done for the functions describing the structure's dynamics. The last ones are obtained using graph theory and the Orthogonality principle [8], [5]. Thé energy conservation law is the base of the method. The energy has two fundamental characteristics - energy flow and energy potential. Thus any physical system is characterized by its general power space which combines countable number of power subspaces in dependence of the different kinds of energy involved in the concrete problem. The parameters (the energy basis) of these power subspaces are specific variables expressing the two basic energy characteristics - the power flow and the power potential. The power flow variables are called "through" ones and the power potential variables - "across" ones. Another important characteristic of every system is its topology. It can be described by graph called general system graph. The component physical characteristics are expressed by relation of its across and through variables, described by mathematical equation which is called terminal equation. Another main class equations are the connection equations. These two classes of equations describe the physical characteristics of the system. And the system topological characteristics are described by another two groups of equations - the cutset and circuit ones. The most general formulation of the Orthogonality principle can be given in the following way:

"If the scalar products of the through and the across variables associated with each edge of a system graph are summed over all edges in the graph then the sum will be zero"

The four groups of equations – the cutset, cirquit, terminal and connection ones, are put in the orthogonality principle and after development in accordance with the method's algorithm the dynamic equations of motion are obtained.

4. Dynamic sensibility

If a force F is supposed to act at the characteristic point as well as a moment M is applied to the last structure link, the dynamic sensibility coefficients and directions can be defined for position and orientation respectively:

$$\alpha_p = (F.\delta R); \quad \alpha_r = (M.\delta\theta); \quad \beta_p = (Fx\delta R); \quad \beta_r = (Mx\delta\theta); \quad (3)$$

They are related to the additional energy, forces and moments have to be compensated to assure optimal energy environment interaction. Let at first the position dynamic sensibility coefficients are considered [12]. The first one - $\alpha_p = (F.\delta R)$, is a scalar and has a dimension of energy. Here the question of maximal and minimal values of α_p is important for practice. It is clear when force direction is perpendicular to some of the ellipsoid axes then the corresponding component of δR disappears and α_p takes lower value. When the kinematic sensibility ellipsoid is one- or two-dimensional it is possible to minimize the coefficient α_p up to zero. Here the role of redundancy is important because it is related to the problem concerning the realization of sensibility directions, following preliminarily given orientations. The coefficient β_p expresses the additional moment caused by the force F in the presence of δR. In the same way the kinematic sensibility ellipsoid is modified, i.e. any of its axis changes its direction in perpendicular plane. All the moments belong to that ellipsoid which will be called dynamic sensibility ellipsoid for position. The dynamic sensibility ellipsoid is three-dimensional when the force direction is non-collinear to its three axes. The most interesting case is when the kinematic ellipsoid is a segment, collinear to the force – then the dynamic ellipsoid disappears, i.e. β_p takes its minimal value – zero. The maximal value of β_p is obtained when the force direction is perpendicular to the biggest kinematic sensibility ellipsoid axis. In the same way the dynamic sensibility ellipsoid for orientation can be defined.

5. Application

The sensibility analysis finds a concrete application during the conception of a mechatronic system, which will be used in the medicine for drilling operation automation.

5.1. KINEMATIC SENSIBILITY PARAMETERS FOR THE MANIPULATIVE STRUCTURE R//T//R

The homomorphism τ_p is described by matrix A and the matrix B_p: The Ker B_p = Ker A is described by two basic eigenvectors:

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad B_p = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad X^{(1)} = [0 \quad 0 \quad 1]^T, \; X^{(2)} = [1 \quad 0 \quad 0]^T \qquad (4)$$

Here Ker A is two-dimensional. And Im A is one-dimensional caused by the eigenvector $X^{(3)} = [0 \quad 1 \quad 0]^T$, corresponding to the positive eigenvalue $\lambda_3 = 1$.

The homomorphism τ_r is described by matrix L and the matrix B_r has the form respectively:

$$L = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \cos(q_1+q_3) & 0 & \cos(q_1+q_3) \end{pmatrix}, B_r = \begin{pmatrix} \cos^2(q_1+q_3) & 0 & \cos^2(q_1+q_3) \\ 0 & 0 & 0 \\ \cos^2(q_1+q_3) & 0 & \cos^2(q_1+q_3) \end{pmatrix} \qquad (5)$$

The Ker B_r = Ker L is described by 2 eigenvectors: $X^{(1)} = [0 \quad 1 \quad 0]^T, X^{(2)} = [-1 \quad 0 \quad 1]^T$

Here Ker L is two-dimensional. And Im L is one-dimensional caused by the eigenvector $X^{(3)} = [1 \quad 0 \quad 1]^T$, corresponding to the positive eigenvalue $\lambda_3 = 2\cos^2(q_1 + q_3)$.

5.2. DYNAMIC EQUATIONS FOR THE MANIPULATIVE STRUCTURE R//T//R

The structure was associated with the graph G_h, consisting of two connected elements G_{tr} and G_r, which is shown on fig.1. The numbers of the graph edges interpret the following variables [5], [12]:

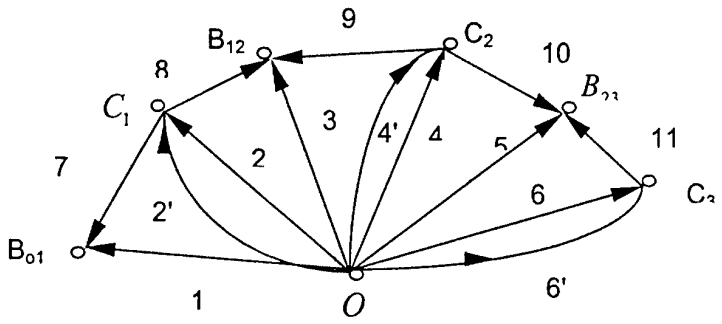


Fig. 1. Graph assigned to the mechanical system

For graph G_{tr}: *Through variables*: D'Alembert forces F_{2k} , (k=1,2,3), associated with the arcs 2k; External forces $F_{(2k)'}$,(k=1,2,3), acting on body k; Forces F_{2k-1}, (k=1,2,3), acting on the terminal points B_{ij} and the origin of the inertial system. Only $F_1 = 0$; Forces F_l^c , (l=2n+1, ... , 4n-1; n=3), presenting the interaction between contiguous bodies.

Across variables: Radius-vectors of the mass-centers and the terminal points B_{ij} are the across variables for all arcs starting from 0 and the local radius-vectors of the points Bij, compared to the mass centers Ci for the remaining arcs. For the formulating tree arcs with numbers from 1 to 2n (n=3) are chosen and all other arcs are chords.

For graph G_r: *Through variables*: D'Alembert torques T_{2k} , associated with arcs 2k (k=1,2,3); External torques $T_{(2k)'}$, (k=1,2,3), acting on body k; Torques T_{2k-1} , (k=1,2,3), for interaction between the terminal points and the inertial beginning; Torques T_l^c, (l=2n1, .. ,4n-1; n=3) for interaction between contiguous bodies.

Across variables: For arcs beginning from 0 across variables are the absolute angular speeds of the bodies, to which points those arcs are directed (points B_{jk} are considered as points of body with number j). Across variables for arcs with number from 7 to 11 describes the relatively angular speeds, as well as across variables with odd numbers are zeros according to the admission for points B_{jk} to be in regarded as appliance to body with number j. For the graph the edges from 1 to 6, which are branches, form a formulation tree and the remaining edges – chords. With the help of four groups of equations from the Orthogonality Principle the differential equations are obtained.

$$A\ddot{q} = B, \quad \begin{bmatrix} I_{33}^{(1)} + I_{33}^{(2)} + I_{33}^{(3)} & 0 & I_{33}^{(3)} \\ 0 & m_2 + m_3 & 0 \\ I_{33}^{(3)} & 0 & I_{33}^{(3)} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} = \begin{bmatrix} T_7 + T_{6'} \\ F_{4'} + F_{6'} + F_9 \\ T_{11} + T_{6'} \end{bmatrix}, \qquad (6)$$

where, m_2 , m_3- the second and third body mass of the mechanical structure;

$J_{33}^{(3)} = \frac{3}{80} m_3 (4r^2 + h)$,

r - the radius of the cone (the cartridge-chamber), h - the height of the cone,

F_9 - translation joint actuator force, T_7 , T_{11}- rotation joint actuator moments,

$F_{4'}$, $F_{6'}$ - external forces, $T_{6'}$ - external moment, q_1, q_2, q_3 - joint variables

The notations are in correspondence with the system graph edges numbering which is used for the differentiation equation derivation.

5.3. DYNAMIC SENSIBILITY PARAMETERS FOR THE MANIPULATIVE STRUCTURE R//T//R.

F, T – main vector and main moment of external action on the last body; λ_p, λ_r – the non-zero (positive) eigenvalues for position and orientation; Following the definitions for dynamic sensibility coefficients, it can be written $\alpha_p = \sqrt{\lambda_p F}$; $\alpha_r = \sqrt{\lambda_r T}$. For the considered structure F consists of gravity force, resistant force (due the contact with the bones) and the internal active force (translation drive).The last two are on the translation direction and the first depends on the concrete drilling position. Both coefficients express the additional power, i.e. energy for unit time the structure needs to compensate the system error, so that it can be minimized in the case when the gravity is perpendicular to the drilling direction. The same can not be said for α_r – its optimization could be done for the reason of λ_r, which is $2\cos^2(q_1 + q_3)$. Next, the redundancy influence here appears to maintain the sum of $q_1 + q_3$ equal or at least close to $\pi/2$. As a result the additional energy for orientation error compensation will be minimal. Finally, the maximal values of α_p and α_r are obtained by evaluation of upper borders of F and T, which could be taken from real experimental results. In the same way the coefficients β_p and β_r can be analyzed for the considered structure. As general, they are related to the additional moments, the system needs to compensate due to corresponding errors. In our case β_p is different from zero only for the gravity force component F^t, i.e. the additional moment appears when F^t is non-collinear to drilling direction. There is dependence between α_p and β_p due to mutual vectors disposition. When α_p increases, β_p decreases at the same time by absolute value. By analogy, the minimal value (zero) of β_r is obtained when the main moment is collinear to the orientation error vector. The maximal values for β_p and β_r also depend on the appropriate evaluations of main force and moment absolute values.

5.4. COMPUTER SIMULATIONS

The sensibility parameters are visualized using MATLAB package. That helps for error distribution and energy deviation analysis in dependence of system state. On the first figure below it is shown that for $(q_1 + q_3) \cong \pi/2$ the kinematic sensibility coefficient for orientation λ_r, is approximately zero and it is denoted by '*' lying in the middle of the sensibility direction for position. The kinematic sensibility directions in the case of position and orientation lie on the same axes i.e. they are collinear what can be seen on the figure.

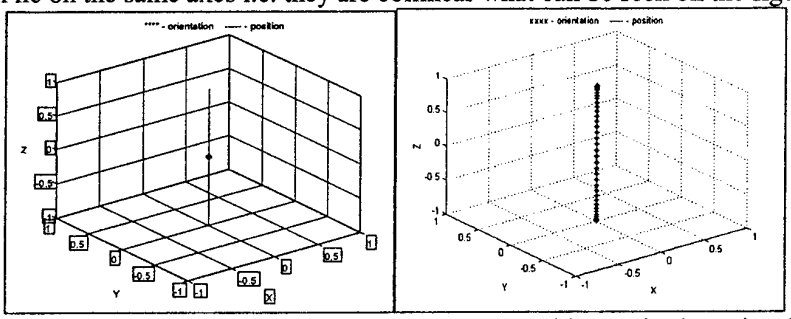


Fig.2. Kinematic sensibility directions in the case of position and orientation for the configurations

$q_1 = 0.7854$ rad, *$q_2 = 95$ mm, $q_3 = 0.8727$ rad;* and *$q_1 = 0$ rad, $q_2 = 5$ mm, $q_3 = 0.7854$ rad*

6. Conclusions

Kinematic sensibility allows the optimization of system accuracy taking into account the system state and redundancy. That gives the possibility to choose the state appropriate for current task execution in accordance with additional criteria. In that sense the sensibility could be considered as generalizing of accuracy. The sensibility ellipsoids content the real deviations inside and give information about their distribution in a limited region.
Dynamic sensibility allows the optimization of energy distribution according to the system state including the external forces and moments interaction in the sense of optimal re-distribution of internal ones. Dynamic sensibility parameters are defined on the subset of kinematic ones, which appear during specific task under force constraints. From the other side that parameters depend also on the external forces and moments variation. For their analysis and interpretation the dynamic model is used. A concrete example of manipulative structure is considered and the obtained results are presented. It is a manipulative structure that will be applied in surgery for drilling operations. For that structure the kinematic and dynamic analyses are done and its dynamic model is represented.

7. References

Journal papers

1. Boiadjiev G., (1996) Kinematic sensible directions of manipulating systems, Proc. of the 6^{th} Int. Symposium on Measirment and Control in Robotics, Brussels, 77-82.
2. Boiadjiev G., Kostadinov K., (1997) Coefficients and Directions of Dynamic Sensibility for Robot Manipulators, Int. Workshop on Adv. Robotics and Inteligent Machins, Manchester, ISSN 11363-2698, PN7, pp.1-11.
3. Vassileva D., Boiadjiev G., (1999) Coeffic. and directions of senibility for orientation of redundant robot-manipulators, SMU, Vol. 3, Drjanovo, ISSN 1310-3946, 1.43-1.49.
4. Vassileva D., Boiadjiev G., (2000) Control of kinematic senibility parameters for planar robot-manipulators with redundancy Edition of Scientific Machine Union, Vol. 3, Drjanovo, ISSN 1310-3946, p. 3.27-3.33, (in Bulgarian).
5. Vassileva D., Hristov K., Boiadjiev G., (2000) Mathematical techniques at Sensibility Analyses of Mechanical Manipulative Structures, Int. Journal "Information - Theories&Applications", vol 7, N 2, ISSN 1310-0513, p. 69-83.
6. Vassileva D., (2000) "Sensibility of Mechatronic Systems with Planar Manipulative Structures", Proc. First Baltic-Bulgarian Conf. on Mechanics, Biomechanics and Bionics, vol. 1, Varna, , ISBN 9984-681-74-2, p. 71-76.
7. Vassileva D., Boiadjiev G, (2000) "Sensibility Analysis of Manipulative Structures with Redundancy", SMU, vol 3 (54), Year VII, 1310-3946 (in Bulgarian), p. 211-216.

Book references

8. Lilov L., Boiadjiev G., (1997), *Dynamics and control of robot-manipulators*, University press "St. K. Ohridski", Sofia, ISBN 954-07-1113-4.
9. Strang G., (1976), *Linear Algebra and its applications*, Acad. Press, New York.
10. Koenig H., Blackwell W., (1962), *Electromech. system theory*, McGraw Hill, London.
11. Andrews G., (1977), *Simulation of multibody systems using the vector- network model. Dyn. Multibody Syst. Symp.*, Munch.

Unpublished papers

12. G. Boiadjiev, R. Kastelov, etc (2002) Robot Application in Medicine for Orthopedic Drilling Manipulation, Mechatronics 2002, Twente, The Netherlands.

STABILITY OF SOLUTIONS OF DIFFERENTIAL EQUATIONS

I.V. BOIKOV, A.I.BOIKOVA
Penza State University,
Krasnay Str.,40, Penza, 440017, Russia

1 Introduction

In the [1] a stability criteria for solutions of systems of differential equations in critical case of one zero root based on analysis of spectrum of the Jacobi matrix for the right-hand side of the equation in a neighborhood of solution is received. This method was generalized in the paper [2] for inverstigation of stability of differential and difference equations in Banach spaces and in all possible critical cases. In the paper [3] the Aizerman's problem was decided for a self-adjoint matrix. In papers [4]-[7] offered criteries of stability of differential equations with lateness, differential equations with small parameters attached to derivative and partial differential equations.

In the second paragraph of the paper we give a short review of these results. New criteria of stability of differential equations in Banach spases are given in the third paragraph.

2 Criteries of stability of solutions of differential and difference equations

2.1 Differential equations

Let us consider a Cauchy problem in a Banach space B:

$$\frac{dx}{dt} = A(x(t)), \tag{2.1}$$

$$x(0) = x_0. \tag{2.2}$$

We assume that: 1) the nonlinear operator A has a continuous first Gateaux derivative; 2) $A(0) = 0$; and 3) the spectrum of the operator $A'(0)$ lies in the left complex half-plane and on the imaginary axis.

Let $\Lambda(A) = lim_{h \downarrow 0} \frac{\|I+hA\|-1}{h}$ be a logarithmic norm of the operator A; moreover, $ReA = A_R = (A + A^*)/2$. Let $R(a,r) = [z \in B : \| z - a \| < r]$ and $S(a,r) = [z \in B : \| z - a \| = r]$.

Theorem 2.1. Let the integral $\int_0^t \Lambda(A'(\phi(\tau)))d\tau$ be strictly negative, i.e. negative and

$$lim_{t\to\infty}\frac{1}{t}\int_0^t \Lambda(A'(\phi(\tau)))d\tau = -\alpha < 0,$$

on an arbitrary differentiable curve $\phi(t)$ in a ball $R(0,r)$ with a sufficiently small radius r. Then the trivial solution of equation (2.1) is stable (asymptotically stable).

Theorem 2.2. Let $\Lambda(\int_0^1 A'(\tau u)d\tau) < 0 (\Lambda(\int_0^1 A'(\tau u)d\tau) \leq -\alpha, \alpha > 0)$ for any $u \neq 0$ belonging to the ball $R(0,r)$ of the space B with a sufficiently small radius r. Then the trivial solution of equation (2.1) is stable (asymptotically stable).

The proofs of Theorem 2.1 and Theorem 2.2 were printed in the [2].

2.2 Difference equations

Let us consider the difference equation

$$x(n+1) = A(x(n)) \tag{2.3}$$

and assume that: 1) the operator A has Gateaux differentiable; 2) $A(0) = 0$; and 3) the spectrum of the operator $A'(0)$ lies inside and on the unit circle with center at the origin.

Theorem 2.3. Let the following conditions hold: 1) the operator $A'(u)$ is completely continuous at all points $u \neq 0$ belonging to the ball $R(0,r)$ with a sufficiently small radius r; and 2) the $s_*(u)-$ numbers of the operators $A'(u)$ are strictly negative (negative and $s_*(u) \leq -\alpha, \alpha > 0$) at the points $u \neq 0$, where $s_*(u)$ is the maximum of the $s-$ numbers of the operators $A'(u)$. Then the trivial solution of equation (2.3) is stable (asymptotically stable).

Theorem 2.4. Let H be a unitary space, and for any $u \neq 0$ let the spectrum of the operator $A'(u)$ consists of distinct eigenvalues with algebraic multiplicity 1 and with absolute values less than one. Moveover, let the eigenvectors corresponding to different eigenvalues be mutually orthogonal. Then the trivial solution of equation (2.3) is stable.

The proofs of Theorem 2.3 and Theorem 2.4 are given in the [3].

2.3 Differential equations with lateness

Let us consider the system of equations

$$\frac{dx_i}{dt} = A_i(x_1(t-h_{i1}(t)),\ldots,x_n(t-h_{in}(t))),\ i = 1,2,\ldots,n. \tag{2.4}$$

Let $t_0 = 0$. We assume that functions $h_{ij}(t)$ are continuous for $t \geq t_0$. Also we assume that $0 \leq max_{ij} \mid h_{ij}(t) \mid \leq H$ for $t_0 \leq t < \infty$. For $t \in [t_0 - H, t_0]$ functions $x_i(t)$ are equal to continuous functions $\phi_i(t), i = 1,2,\ldots,n$. Let $r_0 = max_{1\leq i\leq n} sup_{t\in[t_0-H,t_0]} \mid \phi_i(t) \mid$.

Starting point we write as

$$x^0(t_0+0)=(x_1^0,\dots,x_n^0) \tag{2.5}$$

We will inverstigate the Cauchy task in n dimensional space R_n, with one of the following norms

$$\| x \|_1=[\sum_{k=1}^{n} | x_k |^2]^{1/2}, \| x \|_2= max_{1\le i\le n} | x_i |, \| x \|_3=[\sum_{i=1}^{n} | x_i |].$$

We assume that: 1) $A_i(0,\dots,0)=0, i=1,2,\dots,n$, and 2) functions $A_i(x_1,\dots,x_n), i=1,\dots,n$ are continuous for $x=(x_1,\dots,x_n)\neq(0,\dots,0)$.

Let $x(t)$ be a decision of Cauchy task (2.4)-(2.5).

Let $(s_1(r),\dots,s_n(r))$ be a point that lay on sphere $S(0,r)$. Let us fix an arbitrary matrix $C=[c_{ij}]_{i,j=1,\dots,n}, c_{ij}-const$ with vector $(c_{i1},\dots,c_{in}), i=1,2,\dots,n$ that lay into sphere $S(0,r)$. Let $B(C,r)=[b_{ij}(C,r)]_{i,j=1,2,\dots,n}$, where

$$\mathbf{b_{ij}(C,r)}=\begin{cases}\frac{A_i(0,\dots,0,c_{ij},c_{i,j+1},\dots,c_{in})-A_i(0,\dots,0,c_{i,j+1},\dots,c_{in})}{s_j(r)}, & for\ s_j(r)\neq 0;\\ 0, & for\ s_j(r)=0,\end{cases}$$

$i,j=1,\dots,n$.

Theorem 2.5. Let $C=[c_{ij}]_{i,j=1,\dots,n}$ be an arbitrary nonzero matrix with vectors $(c_{i1},\dots,c_{i,n})\in R(0,r)$. Assume that for $r,r_1>r>r_0$ r_1- arbitrary number $(r_1>r_0))$ the inequality $\Lambda(B(C,r))\le 0(\Lambda(B(C,r))\le-\alpha<0)$ occurs. Then the solution of equation (2.4) is stable (asymptotically stable).

Proof of this Theorem is given in the [5].

2.4 Differential equations with a small parameters attached to derivative

Let us consider a system of differential equations with a small parameter attached to derivative

$$\begin{aligned}\tfrac{dx}{dt}&=Q(x,y,\mu),\\ \mu\tfrac{dy}{dt}&=H(x,y,u,\mu),\end{aligned} \tag{2.6}$$

where $x=(x_1,\dots,x_n); y=(y_1,\dots,y_m); u=(u_1,\dots,u_m); u\in G, \mu(\mu>0)-$ is a small parameter. Vector u is a vector of freedom parameters.

Let $Q(0,y,u,\mu)=0, H(x,0,u,\mu)=0$.

Let $B=(b_1,\dots,b_n), C=(c_1,\dots,c_n), b_i,c_i=const, i=1,\dots,n$ be vectors with arbitrary components.

Let $A(B)=[a_{ij}]_{i,j=1,\dots,n}$, where

$$\mathbf{a_{ij}(B)}=\begin{cases}\frac{Q_i(0,\dots,),0,b_j,\dots,b_n,y,u,\mu)-Q_i(0,\dots,0,b_{j+1},\dots,b_n,y,u,\mu)}{b_j}, & for\ b_j\neq 0;\\ 0\ for\ B_j=0;\end{cases}$$

$i, j = 1, \ldots, n$.

Let $D(C) = [d_{kl}(C)]_{k,l=1,\ldots,n}$, where

$$\mathbf{d_{kl}(C)(B)} = \begin{cases} \frac{H_k(x,0,\ldots,0,c_l,\ldots,c_m,u,\mu) - H_k(x,0,\ldots,0,c_{l+1},\ldots,c_m,u,\mu)}{c-l}, & for\ c_l \neq 0, \\ 0, & for\ c_l = 0, \end{cases}$$

$k, l = 1, \ldots, m$.

Assume that occur the following conditions:

A1) There have place the inequality $\Lambda(A(B)) < 0$ for any nonzero vector $B = (b_1, \ldots, b_n)$ that belong to the sphere $R(0, r) \in R_n$ with a small radius r, for any vector $y = (y_1, \ldots, y_m)$ from the sphere $R(0, r) \in R_m$, for any $u \in G$ and for $0 < \mu \leq \mu_0$.

A2) There have place the inequality $\Lambda(A(B)) < -\alpha, \alpha = const > 0$ with conditions that was formulated in A1.

A3) There have place the inequality $\Lambda(D(C)) < 0)$ for any nonzero vector $C = (c_1, \ldots, c_m)$, that belong to the sphere $R(0, r) \in R_m$ with a small radius r, for any vector $x \in R_n$, for any $u \in G$ and for $0 < \mu \leq \mu_0$.

A4) There have place the inequality $\Lambda(D(C)) < -\beta, \beta = const > 0$ with conditions that was formulated in A3.

Let $\beta_1 = \beta/\mu_0, \gamma = min(\alpha, \beta_1)$.

Theorem 2.6. Let the conditions A1 and A2 occur. Then a domain G is a domain of stability of solutions of the system of equations (2.6).

Theorem 2.7. Let the conditions A3 and A4 occur. Then a solution of the system of equations (2.6) is asymptotically stability uniformly by $\mu, 0 < \mu \leq \mu_0$ and for any $u \in G$.

The proofs of Theorem 2.6 and Theorem 2.7 are given in the [8].

3 Stability of differential equations in Banach spaces

Reviewed in the previous paragraph Theorems are proved with the definition of the following Theorem.

Let us consider nonlinear operational equation in Banach space B

$$\frac{dx}{dt} = A(t, x(t)),\ A(t, 0) \equiv 0. \tag{3.1}$$

Investigate stability of trivial solution of the equation (3.1). Let us set an initial disturbance

$$x(0) = x_0, \quad x_0 \in B \tag{3.2}$$

and consider the Cauchy problem (3.1), (3.2).

Theorem 3.1. Let us assume that in some solid sphere $R(0, \delta)$ of the B space it is satisfied the following condition: for any $(T > 0)$ and any $z \in R(0, \delta)$ it is found such a linear operator $L(T, z)x$, that

1) logarithmic norm of the operator $\Lambda(L(T,z)) < 0 (\Lambda L(T,z) \leq -\alpha, \alpha > 0$

2) for any arbitrary as much as desired small $\epsilon (\epsilon > 0)$ there exists such neighborhood $\delta_1(\epsilon)$ and such an evaluation $\Delta T(\epsilon)$ (for each z is itself), that provided $\|x(t) - z\| \leq \delta_1$ and $t \in [T, T + \Delta T]$ there is valid an inequality

$$\|A(t, x(t)) - L(T, z)x(t)\| < \epsilon.$$

Then a trivial solution of the equation (3.1) is stable (asymptotic stable).

Proof. We lead the proof by contradiction. Let be at the moment $_0$ a trajectory $x(t)$ of Cauchy task (3.1), (3.2) leaves the sphere $S(0, \delta) (\|x_0\| < \delta)$ passing through the point $z \in B$.

By theorem conditions there is found a linear operator $L(T_0, z)$ satisfying the conditions 1), 2). Let us present the equation (3.1) as the form of

$$\frac{dx}{dt} = L(T_0, z)x + F(x), \tag{3.3}$$

here $F(x) = A(t, x(t)) - L(T_0, z)$.

As $z \neq 0$, from the condition 2) of the theorem follows that it will be found such a time interval ΔT^* that providing $t \in [T_0, T_1]$ $T_1 = T_0 + \Delta T^*$

$$\|F(x)\| \leq \epsilon \|x(t)\|. \tag{3.4}$$

Solution of the equation (3.3) for $t \geq T$ there can be presented as the form of

$$x(t) = e^{L(T_0,z)(t-T_0)}x(T_0) + \int_{T_0}^{t} e^{L(T_0,z)(t-\tau)}F(\tau)d\tau. \tag{3.5}$$

Passage in the (3.5) to the norm and taking account of the inequality (3.4) in the time interval $[T_0, T_1]$ we arrive at inequality

$$\|x(t)\| \leq e^{-\alpha(t-T_0)}\|x(T_0)\| + \epsilon \int_{T}^{t} e^{-\alpha(t-\tau)}\|x(\tau)\|d\tau. \tag{3.6}$$

Using standart methods we have in the time interval $[T_0, T_1]$

$$\|x(t)\| \leq e^{-(\alpha-\epsilon)(t-T_0)}\|x(T_0)\|. \tag{3.7}$$

Therefore the trajectory $x(t)$ does not leave the sphere $S(0, \delta)$ and stability is proved.

Let us prove asymptotic stability. By anology with led arguments above we build a sequence of points $T_0, T_1, \ldots, T_n, \cdots$ such that

$$\|x(T_{k+1})\| \leq e^{-(\alpha-\epsilon)(T_{k+1}-T_k)}\|x(T_k)\|. \tag{3.8}$$

There are two possibilities: 1) $\lim\limits_{n\to\infty} T_n = T^*$, 2) $\lim\limits_{n\to\infty} T_n = \infty$.

In the first place passing in the (3.8) to limit providing $k \to \infty$, we have

$$\|x(T^*)\| \leq e^{-(\alpha-\epsilon)(T^*-T_0)}\|x(T_0)\|.$$

It follows that $x(T^*) = 0$. Indeed provided $x(T^*) \neq 0$, then by theorem conditions the trajectory $x(t)$ exists for any $t \geq 0$. Therefore is exists providing $t \geq T^*$. Having taken T^* as initial approximation and having done over again we arrive at contrudiction. Therefore $x(T^*) = 0$.

In the second place tending $x(t)$ to 0 providing $t \to \infty$ is obvious. The theorem is proved.

Remark 1. In case investigation of stability is conducted in Hilbert space, the condition 1) can be changed to the following:

1) $Re(L(T,z)) < 0 (Re(L(T,z)) \leq -\alpha, \alpha > 0)$.

Similar statements are correct and for difference equations.

The paper is supported by Russian Humanities Science Fund (grant 01-02-00147a).

References

[1] Krasovskii, N.N. (1955) Stability of Motions in Critical Case of One Zero Root,*Matematicheskii sbornik*, **Vol.37, N1**,pp.83-88 /In Russian/

[2] Boikov, I.V. (1991) On Stability of Solutions of Differential and Difference Equations in Critical Cases, *Soviet Math. Dokl.*, **Vol.42, N2**, pp.630-632

[3] Boikov, I.V. (1994) Aizerman's Problem, *Prikl. Math. Mekh.* **Vol.58, N4**, pp.52-55

[4] Boikov, I.V. (1997) The Stability of the Motion of a System with Afteraffect, *J. Appl. Maths. Mechs.*, **Vol.3 N61**, pp.385-389

[5] Boikov, I.V. (1998) About Stability of Solution of Differential Equations with Aftereffect, *Differ. equations*, **Vol.8, N34**

[6] Boikov, I.V. (1998) About Definitions of Stability Domeins of System of Differential Equations with Small Parameters for Derivatives, *Automatics and Telemechanics*, **Vol.6**, pp.88-96

[7] Boikov, I.V. (2001) About Definitions Domains of Stability for Some Classes of Non-Linear Equations with Distributed Parameters, *Automatics and Telemechanics*, **Vol.1**, pp.40-49

[8] Poo, T. (1997) *Nonlinear economical dymanocs*. Springer-Verlag. Berlin, Heidelberg.

Simulation of an Active Control System in a Hot-Dip Galvanizing Line

O. Brüls*, J.-C. Golinval

Département Aérospatiale, Mécanique et Matériaux, University of Liège
**Corresponding author. E-mail: o.bruls@ulg.ac.be*

Abstract

This paper concerns the modeling and the integrated numerical simulation of a flexible mechanism subject to the action of a digital control system. A general method is proposed, based on the formalism of flexible multibody systems (MBS) using the Finite Element Method (FEM). Nonlinear effects in the mechanical structure or in the control system can be taken into account. The numerical simulation tool is applied to design an active control system in a hot-dip galvanizing line, which aims at reducing the vibrations of the steel strip.

Keywords: Flexible Multibody Systems, Active Control

1 Introduction

Lately, numerous investigations appeared in control of flexible mechanisms such as flexible manipulators, high precision machine tools, vehicles and foldable structures [5]. The first task to design a control system is to establish the control law defining the relationship between its inputs and its outputs. At this stage, a simple model of the mechanism is required and many dynamical effects are neglected or roughly estimated. Once the control law is defined, the designers try to build a control system composed of actuators, sensors and controllers which will be able to realize the input-output relationship.
However, the actual control system never matches exactly the theoretical control law. Will the actual control system be efficient on the actual mechanism ? Is the control system really optimal for the application ? Often, the answers are obtained through experimental testing and trial-error adjusting of the controller parameters. Instead, designers would prefer to simulate the whole mechatronic system, including the structure, the controller, the sensors and the actuators, using the most rigorous dynamical model as possible. Many standard simulation tools are available in both fields of flexible multibody systems and control systems. But yet, these software packages are usually not able to consider simultaneously the structural behaviour and the control system without lost of generality. The purpose of this paper is to describe and illustrate a general method for the simulation of mechatronic systems.
The mechanical model is built in the formalism of flexible multibody systems, using the Finite Element Method [2]. This formalism, implemented in the MECANO computer code [4], accounts for nonlinear structural flexibility and large displacements. The model of a digital control system is introduced into the simulation as a FORTRAN routine called at each sampling instant. This quite general approach allows to deal with nonlinear effects either in the mechanical structure or in the control system.
The simulation tool is applied to design an active control system in a hot-dip galvanizing line, which aims at reducing the vibrations of the steel strip. The number of actuators and their configuration is defined on the basis of the simulation results.
The time-integration algorithm is presented in section 2 and the design of the active control system for the galvanizing line is described in section 3.

2 Simulation of mechatronic systems

2.1 Multibody dynamics

The purpose of this paragraph is to recall some concepts of multibody dynamics. The finite element methodology is adopted so that the motion is directly referred to the inertial frame. The mechanical system is made of structural components connected through various kinds of kinematic joints.
Applying the Lagrangian multipliers method to the Hamilton principle leads to a system of Differential Algebraic Equations (DAE) of general form [2] :

$$\begin{cases} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{B}^T\lambda - \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{0} \\ \boldsymbol{\Phi}(\mathbf{q}, t) = \mathbf{0} \end{cases} \tag{2.1}$$

where one defines t, the time; $\mathbf{M}$, the mass matrix describing the inertia terms proportional to acceleration; $\mathbf{q}$, the generalized degrees of freedom of the system; $\mathbf{g}$, the sum of external, internal and complementary inertia forces; $\boldsymbol{\Phi}$, the set of holonomic kinematic constraints; λ, the set of Lagrangian multipliers and $\mathbf{B}$, the matrix of constraint gradients.
The first set of equations describes the dynamic equilibrium of the system and the second one represents the holonomic kinematic constraints.
The equations (2.1) may be solved through time domain numerical integration with the well known Newmark α-family of implicit algorithms. Assuming that the solution is known at time t_n, the unknowns of the problem are $\mathbf{q}_{n+1}$, $\dot{\mathbf{q}}_{n+1}$, $\ddot{\mathbf{q}}_{n+1}$ and λ_{n+1} at time $t_{n+1} = t_n + h$, where h is the time step. The Newton-Raphson iterative procedure is applied to converge to the solution by successive linearizations of the equations.

2.2 Simulation of Mechatronic Systems

A sampled-data control system with a sampling period T can be modelled by the following state equations:

$$\mathbf{x}_{i+1} = \mathbf{f}_u(\mathbf{x}_i, \mathbf{u}_i, t_i) \tag{2.2}$$

$$\mathbf{f}_{i+1} = \mathbf{f}_o(\mathbf{x}_i, \mathbf{u}_i, t_i) \tag{2.3}$$

where the subscript i denotes the i^{th} sampling instant, $\mathbf{u}_i$ the vector of the inputs, $\mathbf{x}_{i+1}$ the state vector of the control system, and $\mathbf{f}_{i+1}$ the vector of the outputs applied to the mechanism during the time interval $[t_i, t_{i+1}]$. $\mathbf{f}_u$ and $\mathbf{f}_o$ are respectively the update and output functions. In our case, the input data are measured on the mechanical system, and thus are related with its generalized coordinates: $\mathbf{u} = \mathbf{u}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$. The action of the control system on the structure modifies its dynamic equilibrium :

$$\begin{cases} \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{B}^T\lambda - \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathbf{D}\mathbf{f}_{i+1} & \forall t \in [t_i, t_{i+1}] \\ \boldsymbol{\Phi}(\mathbf{q}, t) = \mathbf{0} \end{cases} \tag{2.4}$$

where $\mathbf{D}$ is the influence matrix of the control forces on the generalized coordinates, which is assumed to be constant. The integration algorithm of equations (2.4) is illustrated in figure 1. The time step h is a divisor of the sampling period T. Inside each sampling period, the time integration of the mechanical equations is performed taking into account a constant vector $\mathbf{f}$. At the sampling instants, the control routine updates the control forces as well as the state variables. In most cases, the dynamics of the control system is faster than the dynamics of the mechanism and a reasonable choice for the time step is $h = T$.

3 Application of the simulation tool

3.1 Galvanizing process

Figure 2 illustrates a continuous hot-dip galvanizing line. The steel strip, of the order of 1 m wide by 1 mm thick, is preheated and passed at the speed of about 1 m/s through a pot of molten zinc. A zinc film is entrained onto the strip as it emerges from the pot. The deposited film solidifies while the strip

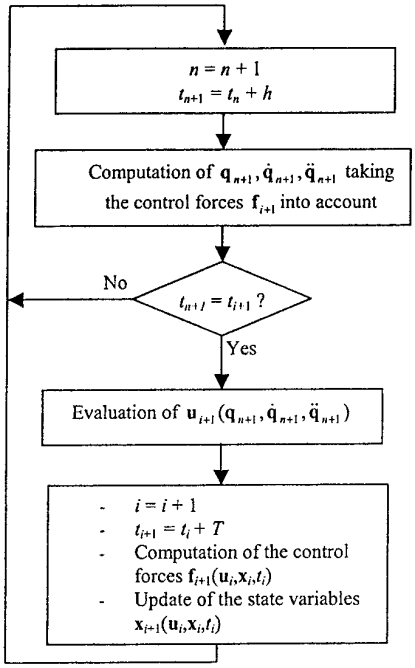


Figure 1: Numerical integration algorithm for a mechanical system subject to the action of a digital controller.

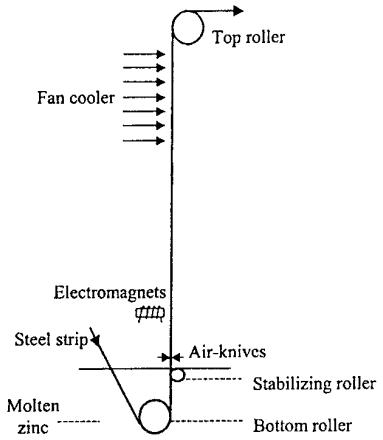


Figure 2: Galvanizing line.

Flexion (Hz)	Torsion (Hz)
0.55	0.56
1.10	1.11
1.66	1.67
2.21	2.23
...	...

Table 1: Natural frequencies of the structure.

runs vertically upwards. After the top roller, the finished product is guided to a delivery section where it is coiled and cut. The distance from the stabilizing roller to the top roller is of the order of 50 m.
Accurate control of the amount of solidified deposit has a great commercial issue: overdeposition results in excessive use of zinc which increases the production costs; underdeposition results in an unsatisfactory product. Air-knives, consisting of a pair of nozzles, regulate the zinc thickness. However, the vibration movement of the steel strip in front of the air-knives leads to variations in the amount of deposit.
Our purpose is to design a collocated active control system able to reduce those vibrations. Based on the information received by a sensor, a digital controller drives an electromagnet acting on the strip. Several independent sensor-actuator pairs can thus be installed.
In the following paragraphs, a mechanical model is first established. Then the design and the modeling of the control system is described. Finally, the simulation tool which has been presented above is used to estimate the performances of the system.

3.2 Mechanical modeling

Although the steel strip is prestressed, the structure remains very flexible and the mechanical excitation induced by the fan cooler causes a high vibration level. This section aims at constructing a reliable mechanical model able to capture all the significant effects.

Basic model The steel strip may be assumed to be fixed at the stabilizing roller and at the top roller. It is modeled with shell finite elements which allows to take the prestressing effect and the gravity field into account. As the strip bends, its extension produces modifications of the stresses inside the structure, which influences its stiffness. This nonlinear phenomenon is well known as geometrical stiffening and was considered in a preliminary model. But the results showed that the stress modifications remain small, so that the geometrical stiffening can be neglected and a linear model is sufficient to describe the dynamic behaviour of the steel strip.
The pressure field produced by the fan cooler is modeled as a white noise excitation in the frequency range from 0.2 Hz to 10 Hz, as suggested by experimental data. This excitation appears to be spatially uncorrelated. However, to avoid the definition of a time domain excitation function at each node of the finite element model, the excitation zone may be decomposed into several independent zones in which the nodes are simultaneously excited.
The natural frequencies were computed for the linear model and the results are presented in Table 1. The eigen-frequencies of the torsion modes almost match the eigen-frequencies of the flexion modes.

Speed of the steel strip The vertical motion of the steel strip during the process may affect the vibrations. A two-dimensional finite element model of the moving steel strip has been developed to study this phenomenon. The galvanizing line has been replaced by a line enclosing the stabilizing roller and the top roller as shown in figure 3. Despite numerical difficulties encountered in the elaboration of this model, the eigen-frequencies have been computed for increasing values of the vertical speed. For speed values up to 15 m/s, the natural frequencies remain almost unaffected.
As a conclusion, we can assume that the steel strip is motionless and that both the stabilizing roller and the top roller are fixed.

Model reduction All nonlinear effects were found to be negligible in the structure. Thus, assuming a linear behaviour, the Craig-Bampton substructuring method can be used to build a reduced model containing less degrees of freedom [2]. This method requires the partitioning of the initial degrees of

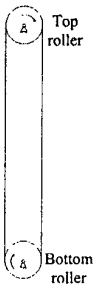


Figure 3: Two-dimensional finite element model of the moving steel strip.

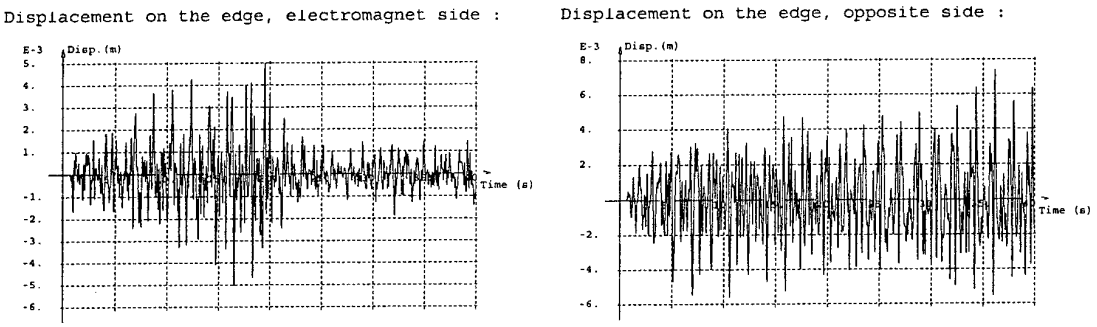


Figure 4: Displacements in front of the air-knives when a single actuator is located on one edge of the steel strip; the active control system is turned on at time $t = 20s$ (sampling period : $T = 0.5$ ms, gain : $g = 100$)

freedom into two groups : the boundary degrees of freedom, which will be retained, and the internal degrees of freedom which will not appear explicitly in the reduced model and are considered as free. The movement of the structure is described as the superposition of constrained modes describing the static behaviour of the boundaries and a few clamped vibration modes obtained when fixing the boundary.
The degrees of freedom situated on the rollers, in the excitation zone, in front of the electromagnets and in front of the air-knives are defined as boundary degrees of freedom. To cover the frequency range of the excitation (0.2 Hz - 10 Hz), 50 clamped vibration modes have been kept. As the initial model contains 2400 degrees of freedom, the reduced one contains only 300 degrees of freedom so that the computation time decreases by a factor of 3.

3.3 Active control system

This paragraph concerns the design and the modeling of the active control system when a single actuator acts on the structure (single input - single output system). In the case of a multiple input - multiple output system, a control routine has to be defined for each independent sensor/actuator pair.

Control law Design methods for active control system are extensively described in the reference [3]. This paragraph presents the results of the design procedure.
A collocated configuration of the actuator and the sensor is chosen in order to maximise the robustness. The active damping control law is a direct velocity feedback :

$$f_j^d = -g\dot{q}_j \tag{3.5}$$

The desired force f_j^d applied on node j is proportional and opposite to the measured velocity $\dot{q}_j$, which

guarantees energy dissipation and unconditional stability. This control law is thus stabilizing for any flexible structure. No matter the dimensions of the steel strip, all vibration modes will be damped. The gain g, which defines the impact of the control system on the structure, has to be carefully optimized.

Actuator placement The performance of the active control depends on the position of the actuator on the structure. A method developed by Gawronski [1] has been applied to find the best location of the actuator. The detailed description of this method is beyond the scope of this paper. In brief, accounting for technological constraints, the optimization yields the following conclusion: the actuator should be placed 3.5 m above the stabilizing roller, on the edge of the strip in order to control both flexion and torsion modes.

Actuator modeling The controller drives the electrical current i in the electromagnet. But, the relation between the electrical current and the force f_j applied on the steel strip is highly nonlinear and dependent on the air gap e. An analytical expression of the relation $f_j(i, e)$ has been established to fit experimental data. This relation is the non-linear model of the actuator.

Description of the control routine The control routine receives the input vector $\mathbf{u}$ containing two components: q_j and $\dot{q}_j$. First, it computes f_j^d according to the control law (3.5). Then, the desired electrical current i is adequately estimated. The value of q_j is used to compute the air gap e. Finally, the actuator model $f_j(i, e)$ defines the force $\mathbf{f}$ applied on the structure.

3.4 Results

A parametric study has been led for several configurations of the actuators and several dimensions of the steel strip. For the sake of conciseness, the detailed results are not presented here. Figure 4 illustrates the results obtained with mean dimensions of the strip and a single actuator placed on the edge, 3.5 m above the stabilizing roller. After 20 seconds, the control system is turned on and the vibrations are efficiently attenuated on the actuator side, but not on the other side. Better performances are observed with more actuators: three actuators are able to reduce efficiently the vibration level in front of the air-knives.

4 Conclusion

The general simulation tool presented in this paper is adapted for the simulation of any flexible mechanism subject to the action of any digital control system. It turned out to be really helpful for the design of an active control system in a hot-dip galvanizing line. However, one difficulty of the method is its huge computational load.
This tool may be extended to many other kinds of applications as the modeling of machine tools, flexible manipulators, foldable structures, vehicles...

Acknowledgements

M. Brüls is supported by a grant from the Belgian National Fund for Scientific Research (FNRS) which is gratefully acknowledged.

References

[1] W.K. Gawronski, *Dynamics and Control of Structures - A Modal Approach*, Springer,1998.

[2] M. Géradin, A. Cardona, *Flexible Multibody Dynamics : A Finite Element Approach*, John Wiley & Sons,1997.

[3] A. Preumont, *Vibration Control of Active Structures*, Presses Universitaires de Bruxelles a.s.b.l.,1996.

[4] SAMTECH, *Manuel d'utilisation de SAMCEF, v8.0*, Liège, Belgique,1999.

[5] H. Van Brussel, P. Sas, I. Németh, P. De Fonseca, P. Van den Braembussche, Towards a Mechatronic Compiler, *IEEE/ASME Transactions on Mechatronics* **6** no. 1 (2001) 90–105.

Coupled Multibody-Aerodynamic Simulation of High-Speed Trains Manoeuvres

Antonio Carrarini (antonio.carrarini@dlr.de)
DLR – German Aerospace Center, Vehicle System Dynamics Group
P.O.Box 1116, D–82230 Wessling, Germany

Abstract. In this paper the effects of unsteady aerodynamic loads on the driving dynamics of high speed trains during passing manoeuvres in absence of cross wind have been investigated. To this end a co-simulation MBS/CFD was implemented. A linear aerodynamic model, the panel method, was applied to the computation of the unsteady flow around the driving trailers for the examined manoeuvres. The multibody simulation program SIMPACK simulated the dynamic response of the vehicles to the resulting aerodynamic loads.

Keywords: Multibody dynamics, railway aerodynamics, unsteady aerodynamics, co-simulation, coupled systems

1. Introduction

Rapidly growing operative speeds together with the cut off of leading car's weight, due to light construction and to the distribution of the traction units along the whole train, let today's trains be very sensitive to aerodynamic loads. For example, the driving trailers of many recent high speed trains are precautionary ballasted in order to reduce their aerodynamic sensitivity. The response of vehicles to steady and especially unsteady loads has thus to be carefully investigated to ensure the safety of railway operations under extreme aerodynamic conditions [1].

The most general way to include aerodynamic effects in a multibody system is the coupling of the multibody system code with a solver from computational fluid dynamics (CFD), see [2, 3]. Such *partitioned* approach, which is called *co-simulation* or *simulator coupling* when the coupled codes remain unchanged and completely stand-alone and communicate only through appropriate interfaces at discrete time points, see [4], is capable to describe virtually every *unsteady* aerodynamic phenomenon and to take into account the reciprocal interaction between mechanical and aerodynamical system.

A new application field for this coupled approach is the behavior of ground vehicles under unsteady aerodynamic loads, for example due to the interaction with other vehicles (*interference*), see [5, 6]. Such problems can not be handled by the conventional approach based on aerodynamic coefficients. The typical case of two high speed trains passing by each other is presented below.

It must be mentioned that the methods of the multibody dynamics and their implementation in simulation software offer very efficient tools for the analysis of the dynamical behavior of railway vehicles. On the contrary the description of unsteady aerodynamic loads through CFD methods or wind

tunnel experiments can be still achieved only with great efforts and high costs, in most cases with poor accuracy.

2. Basic principle of the co-simulation

The modular structure of coupled problems may be adopted in the simulation using for each subsystem its own simulation tool for model setup and time integration [7]. Well established standard software tools are used for the individual subsystems. In this way the subsystems are integrated by *different* time integration methods such that each of these methods can be tailored to the solution behavior of the corresponding subsystem.

The communication between subsystems is restricted to discrete synchronization points T_n. In each subsystem all necessary information from other subsystems can be provided by interpolation or — if data for interpolation are not yet available — by extrapolation from $t \leq T_n$ to the actual *macro step* $T_n \rightarrow T_{n+1}$. But in many cases it is sufficient to keep the value of the coupling variables from the other subsystems constant during the whole macro step $T_n \rightarrow T_{n+1}$. The latter is the usual approach used by the multibody system tool during the co-simulation.

Co-simulation techniques are convenient but they may suffer from numerical instability. Furthermore, interpolation and extrapolation introduce additional discretization errors. In most standard applications stability and accuracy is guaranteed if the *macro step size* $H := T_{n+1} - T_n$ is sufficiently small.

For certain classes of coupled problems the instability phenomenon has been analyzed in great detail. Several modifications of the co-simulation techniques help to improve its stability, accuracy and robustness also for larger macro step sizes [8].

3. Formulation of the coupled problem

3.1. Multibody System

The classical topic of interest in multibody dynamics are systems of rigid bodies being connected by joints and force elements like springs and dampers [9]. The equations of motion are given by

$$\mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}}(t) = \mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}}, \lambda) - \mathbf{G}^T(t, \mathbf{q})\,\lambda\,, \tag{1a}$$

$$0 = \mathbf{g}(t, \mathbf{q}) \tag{1b}$$

with $\mathbf{q}$ denoting the position coordinates of all bodies. $\mathbf{M}(\mathbf{q})$ is the generalized mass matrix and $\mathbf{f}$ the vector of applied forces. Joints decrease the number of degrees of freedom in the system and may result in constraints (1b) that are coupled to the dynamical equations (1a) by constraint forces $-\mathbf{G}^T\lambda$ with Lagrange multipliers λ and $\mathbf{G}(t, \mathbf{q}) := (\partial \mathbf{g}/\partial \mathbf{q})(t, \mathbf{q})$. Very efficient numerical

methods for the evaluation and for the time integration of (1) have been developed and implemented in industrial multibody simulation tools like ADAMS, SIMPACK or DADS, see [10, 11].

Already in the early days of multibody dynamics these methods have been extended to more general mechanical systems that contain e. g. flexible bodies or force elements with internal dynamics. On the contrary the extension of the simulation scenarios through co-simulation is a recent development which is still in progress.

3.2. Aerodynamic System

The flow around high speed trains in absence of cross wind can be assumed to be inviscid and irrotational, leading to a linear aerodynamic model. Such a flow model is called *potential* flow and is widely used in aircraft aerodynamics but also in railway aerodynamics when cross wind has not to be considered. Its discretized numerical formulations, the *panel methods* [12, 13], lead to small computational effort and other benefits compared to nonlinear aerodynamic models.

A potential flow can be merely described by the Laplace equation by introducing a scalar field function Φ:

$$\nabla^2 \Phi = 0 \tag{2}$$

whereby potential function and velocity field are directly connected:

$$\mathbf{u} = \nabla \Phi \,. \tag{3}$$

Eq. (2) must be completed with some boundary conditions which are the physical interface between multibody and aerodynamical system. Such conditions are presented in the next section.

3.3. Coupled System

The boundary condition for the flow only requires that the normal component of the relative velocity on the vehicles walls $\partial\Omega_V$ vanishes, i. e. that the normal component of the absolute velocity $\mathbf{u}$ is equal to the velocity of the wall $\mathbf{v}$:

$$\nabla \Phi \cdot \mathbf{n} = -\mathbf{v}(\dot{\mathbf{q}}) \cdot \mathbf{n} \qquad \text{on } \partial\Omega_V \tag{4}$$

which shows that the potential Φ must depend on the velocity of the vehicles $\dot{\mathbf{q}}$.

Using Green's formula Eq. (2) can be rearranged to obtain an expression for the potential Φ as integral on the vehicles walls $\partial\Omega_V$ of a *source* distribution σ divided by the module of the position vector $\mathbf{r}$. A *doublet* distribution, which compares in the general formulation, is not necessary for the case of ground vehicles because no special conditions, such as the Kutta–condition, have to be satisfied.

Since $\partial\Omega_V$ depends on the vehicles position, Φ depends on $\mathbf{q}$ as well:

$$\Phi(\mathbf{r}, \mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{4\pi} \int_{\partial\Omega_V} \sigma \frac{1}{|\mathbf{r}|} \mathrm{d}s \,. \tag{5}$$

The source distribution σ on $\partial\Omega_V$ is unknown and has to be determined using the boundary condition (4). When σ has been computed, Φ and $\mathbf{u}$ can be derived using (5) and (3).

The Bernoulli Equation can now be applied to obtain the pressure field:

$$\frac{\partial\Phi}{\partial t} + \frac{|\mathbf{u}|^2}{2} + \frac{p}{\rho} = \frac{p_\infty}{\rho} = \text{const} \quad \Rightarrow \quad p(\mathbf{r}, \mathbf{q}, \dot{\mathbf{q}}, t) \,. \tag{6}$$

It is finally possible to compute the resulting flow force $\mathbf{L}_f$ and torque $\mathbf{M}_f$ related to the origin O:

$$\mathbf{L}_f(\mathbf{q}, \dot{\mathbf{q}}, t) \;=\; -\int_{\partial\Omega_V} p \cdot \mathbf{n} \, \mathrm{d}s \,, \tag{7a}$$

$$\mathbf{M}_f(\mathbf{q}, \dot{\mathbf{q}}, t) \;=\; -\int_{\partial\Omega_V} \mathbf{r} \times p \cdot \mathbf{n} \, \mathrm{d}s \tag{7b}$$

which couple the flow equations (2) and (4) with the multibody system equations (1).

The used panel method adopts a discretization of the surface integral in (5). The finite surface elements are called *panels* and on each of them the source distribution σ_i is supposed to be constant. The boundary condition (4) leads to an algebraic linear system whose unknown vector is the discrete source distribution σ_i and whose dimension is thus the number of panels. Eq. (6) has also to be discretized: pressure distribution and forces (7) can be finally obtained on a discrete time axis.

In order to minimize the computational effort the number of "aerodynamic" time steps must be minimized, as each of these time steps required for a usual configuration about 15 minutes. The panel method is capable of very large time steps compared to the multibody system part. Furthermore, the flow and driving dynamics are quite weakly coupled. For these reasons a co-simulation technique has been implemented. In each macro step $T_n \rightarrow T_{n+1}$ Eq. (6) is discretized once using the macro step size H as time step. The flow field is thus resolved only at the synchronization points T_n and kept frozen between them. The multibody system part of the coupled problem is integrated by standard techniques from multibody dynamics with step size and order control. In this way about 30 macro steps are necessary for the simulation of a typical manoeuvre.

4. Results

The simulation of a wide range of typical driving manoeuvres (passing on open track and at tunnel entrance, tunnel run-in and run-out, etc.) have been

performed. Results can be examined in many ways using different criteria but none of them can be definitively chosen as representative, as each railway company uses its own methods to estimate aerodynamic sensitivity. In the following only the lateral displacement of the leading wheelset for the case of two ICE trains passing by each other at the same speed on open track is reported.

Results show that, even if the aerodynamic forces grow to the square of the driving velocity, the response of the system has only a linear dependence on the velocity, see Fig. 1. As a consequence not exclusively very high driving speeds are critical. Results plotted in Fig. 1 refer to trains driving on a perfectly straight and plane track without rail excitations (ideal case); the values are therefore relative small.

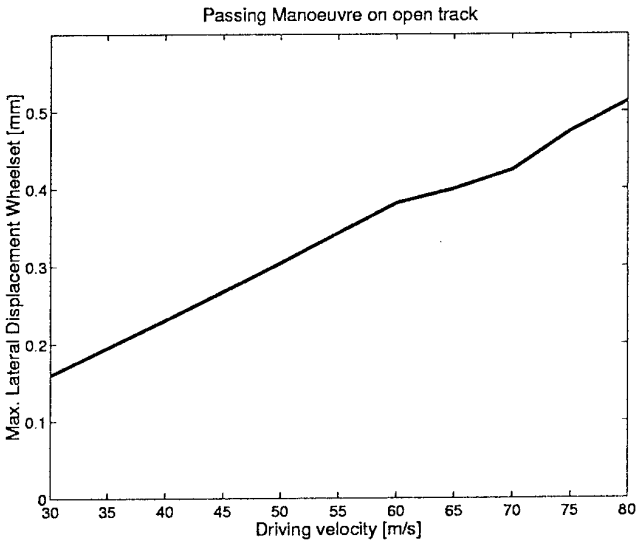


Figure 1. Maximal wheelset's lateral displacement during a passing manoeuvre on open track (ideal case).

From the simulations also emerged that the presence of little disturbances can amplify the dynamical response of the vehicles. Fig. 2 shows a typical situation: a small, low frequency perturbation, which could be caused by cross wind or track irregularities, lets the maximal displacement of the wheelset reach much larger values than in the ideal case.

Using the new simulation tool it was also possible to point out that, whereas the unsteady aerodynamic loads can exert a very large influence on the driving dynamics, the effects of the induced vehicle motion on the surrounding flow is of some influence only when the fundamental frequency of the excitation approaches the lowest natural frequencies of the car. In the case of symmetrical passing manoeuvres such condition is satisfied only at very small driving velocities, the influence can be thus usually neglected.

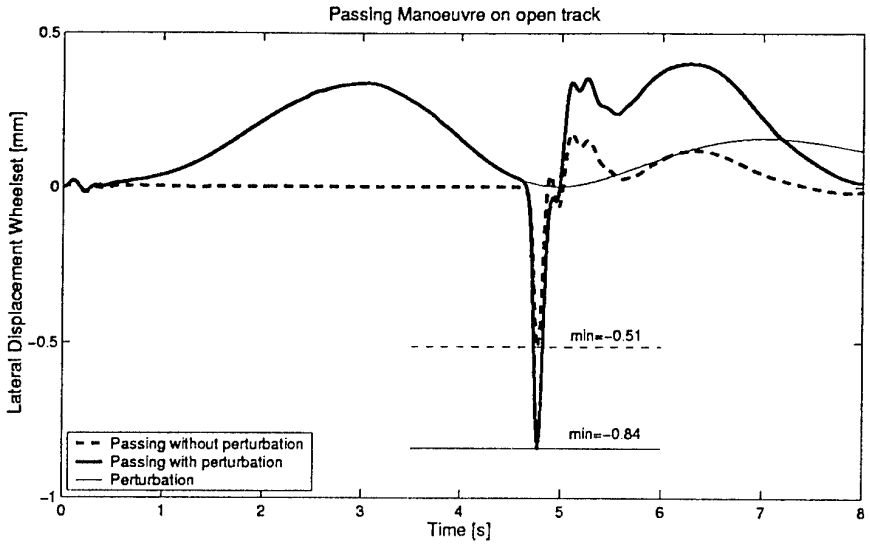


Figure 2. Effect of a small perturbation on wheelset's lateral dynamics during a symmetrical passing manoeuvre on open track at 80 m/s.

References

1. S. Lippert. On side wind stability of trains. Technical Report TRITA - FKT Report 1999:38, Royal Insitute of Technology – Railway Technology, Stockholm, 1999.
2. H.G. Matthies and J. Steindorf. Efficient iteration schemes for nonlinear fluid-structure interaction problems. In B.H.V. Topping, editor, *Computational Mechanics: Techniques and Developments*, pages 263–267. Civil-Comp Press, Edinburgh, 2000.
3. W. Rumold. Modeling and simulation of vehicle carring liquid cargo. *Multibody System Dynamics*, 5:351–374, 2001.
4. A. Veitl, T. Gordon, et al. Methodologies for coupling simulation models and codes in mechatronic system analysis and design. In *16th IAVSD Symposium*, Pretoria, South Africa, 1999. Supplement to Vehicle Sytem Dynamics, Vol. 33, 2000.
5. S. Yamamoto et al. Aerodynamic influenece of a passing vehicle on the stability of the other vehicles. *JSAE Review*, 18:39–44, 1997.
6. K. Pahlke. Application of the standard aeronautical CFD method FLOWer to trains passing on open track. In *TRANSAERO Symposium, Paris*, 1999.
7. M. Arnold, C. Carrarini, A. Heckmann, and G. Hippmann. Modular dynamical simulation of mechatronic and coupled systems. In H.A. Mang, F.G. Rammerstorfer, and J. Eberhardsteiner, editors, *Proc. of WCCM V, Fifth World Congress on Computational Mechanics, July 7–12, 2002, Vienna, Austria*, 2002.
8. R. Kübler and W. Schiehlen. Two methods of simulator coupling. *Mathematical and Computer Modelling of Dynamical Systems*, 6:93–113, 2000.
9. R.E. Roberson and R. Schwertassek. *Dynamics of Multibody Systems*. Springer-Verlag, Berlin Heidelberg New York, 1988.
10. W.O. Schiehlen, editor. *Multibody Systems Handbook*. Springer–Verlag, Berlin Heidelberg New York, 1990.
11. W. Rulka. *Effiziente Simulation der Dynamik mechatronischer Systeme für industrielle Anwendungen*. PhD thesis, Vienna University of Technology, Department of Mechanical Engineering, 1998.
12. J.L. Hess and A.M.O. Smith. Calculation of potential flow about arbitrary bodies. *Progress in Aeronautical Sciences*, 8, 1985.
13. L.L. Erickson. Panel Methods - An Introduction. Technical Paper 2995, NASA, 1990.

EFFICIENT SIMULATION OF RIGID-FLEXIBLE MULTIBODY DYNAMICS: SOME IMPLEMENTATIONS AND RESULTS*

O. N. DMITROTCHENKO
Department of Applied Mechanics, Bryansk State Technical University
Bulv. im. 50-letiya Oktyabrya 7, 241035 Bryansk, Russia
don@bitmcnit.bryansk.su

Abstract. Known and modified simulation methods, such as composite and articulated ones, as well as different finite-element discretization methods are presented.

1. Equations of motion

At first the dynamic equations of a flexible body are considered in the form of the so-called [1] *semidiscretized equations*

$$\begin{aligned} M^r w + M^{re}\ddot{u} &= Q^r + R^r, \\ M^{re\,T} w + M^e \ddot{u} &= Q^e + R^e, \end{aligned} \tag{1}$$

where M^r, M^{re} and M^e are the (quasi-)rigid, rigid-elastic and elastic mass matrices; the next three 6-columns relate to motion of the body-fixed *floating reference frame* [2]: linear and angular accelerations $w = (\ddot{r}^T \quad \dot{\omega}^T)^T$, applied and inertial forces and their moments Q^r, reaction forces and moments R^r; finally, Q^e and R^e are the applied/inertial and reaction generalized forces relating to dynamics of the generalized coordinates u defining the deformed state of the body.

Now let the motion of the reference frame origin for each body i be defined by means of a set of generalized coordinates q_i as follows:

$$w_i = \Phi_i \ddot{q}_i + w_i'.$$

Here Φ_i are Jacobian matrices. The second discretization of equations (1) by substitution of w_i and summation over all the bodies in a system lead to the equations

$$\sum_{i=1}^{n} \left(\begin{bmatrix} \Phi_i^T M_i^r \Phi_i & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & \Phi_i^T M_i^{re} \\ M_i^{re\,T} \Phi_i & M_i^e \end{bmatrix} \right) \begin{Bmatrix} \ddot{q}_i \\ \ddot{u}_i \end{Bmatrix} = \sum_{i=1}^{n} \begin{Bmatrix} \Phi_i^T \left(Q_i^r - M_i^r w_i' \right) \\ Q_i^e - M_i^{re\,T} w_i' \end{Bmatrix},$$

or, in a compact form, the *equations of structural dynamics* in generalized coordinates

$$\left(\hat{M}^r + \hat{M}^e \right) \ddot{x} = \hat{Q}. \tag{2}$$

*) Supported by the Russian Foundation for Basic R esearch under the grants 02-01-00364, 02-01-06098 and by the scientific program "Universities of Russia – Basic Research" (UR.04.01.046).

2. Simulation methods

2.1. COMPOSITE BODY METHOD

Effectiveness of simulation of a large system can be estimated for a n-body chain (Figure 1). So, a *direct method* of implementation of the equations (2) is *cubic* in n. That is the computational effort is $O(n^3)$ for the matrix $\hat{M}^r$ and $O(n^2)$ for $\hat{M}^e$, $\hat{Q}$.

The composite body method [3] known for rigid multibody systems allows decreasing the effort to a *quadratic* one: down to $O(n^2)$ and $O(n)$ for the matrices above:

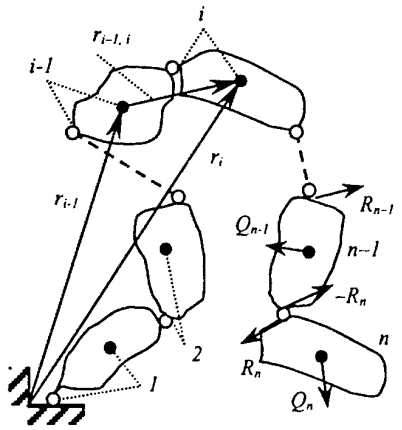


Figure 1. n-body chain

$$\hat{M}^r = \sum_{i=1}^{n}\sum_{j=1}^{n} \Phi_i^{*T} M_{ij}^{*} \Phi_j^{*},$$

$$M_{ij}^{*} = \sum_{k=\max(i,j)}^{n} C_k^T M_k^r C_k, \quad \Phi_i^{*} = C_i^{-1} S_i^{*},$$

$$C_k = \begin{bmatrix} I_{3\times3} & -\tilde{r}_k \\ 0 & I_{3\times3} \end{bmatrix}, \quad S_i^{*} = [0 \ldots S_i \ldots 0].$$

Here r_i is the reference frame origin of the i-th body; S_i is a Jacobian matrix describing the local kinematics of two contiguous bodies:

$$\begin{Bmatrix} \dot{r}_i \\ \omega_i \end{Bmatrix} = \begin{bmatrix} I_{3\times3} & -\tilde{r}_{i-1,i} \\ 0 & I_{3\times3} \end{bmatrix} \begin{Bmatrix} \dot{r}_{i-1} \\ \omega_{i-1} \end{Bmatrix} + S_i \dot{u}_i. \qquad (3)$$

For a rigid body, u_i is a column of local joint coordinates in joint i.

2.2. ARTICULATED BODY METHOD

Now an application of the articulated body method [4,5] for a rigid-flexible multibody system is considered. The method is *linear* in n for a n-body chain because it does not deal with a global mass matrix at all, but uses a recurrent two-step procedure instead in order to eliminate reaction forces from the equations of motion of separate bodies.

Direct motion. Let us consider two end bodies n and $(n-1)$ of the chain in Figure 1.

Let both bodies n and (n–1) be flexible. Equations (1) for body n are

$$M_n^r w_n + M_n^{re} \ddot{u}_n = Q_n^r + R_n, \qquad (4)$$

$$M_n^{re\,T} w_n + M_n^e \ddot{u}_n = Q_n^e. \qquad (5)$$

Body $(n-1)$ is subjected to influence of both R_n and R_{n-1} reaction forces:

$$\begin{aligned} M_{n-1}^r w_{n-1} + M_{n-1}^{re} \ddot{u}_{n-1} &= Q_{n-1}^r + R_{n-1} - C_n^T R_n, \\ M_{n-1}^{re\,T} w_{n-1} + M_{n-1}^e \ddot{u}_{n-1} &= Q_{n-1}^e \qquad - S_n^T R_n. \end{aligned} \qquad (6)$$

Accelerations of the two bodies are coupled analogously to equation (3):

$$w_n = C_n w_{n-1} + S_n \ddot{u}_{n-1} + w_n'. \qquad (7)$$

Substitution of R_n, $\ddot{u}_n$ and w_n from equations (4),(5),(7) eliminates reactions R_n from equations (6).

Let the body n be flexible, but the body (n-1) be rigid. The equation (6) change into

$$M^r_{n-1} w_{n-1} = Q^r_{n-1} + R_{n-1} - C^T_n R_n \,. \tag{8}$$

Due to the *ideal constraints* in joint n, the following condition holds:

$$S^T_n R_n = 0 \,. \tag{9}$$

This also leads to eliminating the reaction forces R_n from equation (8).

Thus, one can turn from body n to $(n-1)$ and go on the process down to body 1.

Reverse motion. Equation (5) written down for body 1 gives the generalized accelerations $\ddot{u}_1$. Further, with the help of kinematical relation (7), the acceleration w_2 of the next body 2 can be found, and the process goes on to the end of the chain.

2.3. ARTICULATED METHOD FOR CONSTRAINED RIGID-BODY SYSTEMS

Despite its high effectiveness, the idea of the latter method does not work if closed kinematical loops exist. In this case, a modification of the articulated body method was proposed [5], which is based on transition to Lagrange multipliers in constraints.

Let us consider a constrained rigid multibody system.

The equation (9) can be solved relative to the reaction forces R_n as follows:

$$R_n = H_n \lambda_n \,, \qquad \text{then} \qquad H^T_n S_n = 0 \,. \tag{10}$$

Here λ_n is a column of Lagrange multipliers (independent reactions), $H_n = \ker S^T_n$.

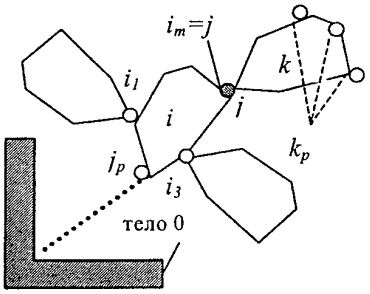


Figure 2. Joint and its environment

Figure 2 shows a joint j of a system and some contiguous bodies. The joint connects two bodies k and i. The previous joint j_p connects body i and the previous body on the path to body 0 (fixed inertial frame). Other joints attached to body i are denoted by i_1, ..., i_m, ...; one of them, e.g. i_m, is joint j, obviously. Joints attached to body k are k_1, ..., k_p, ...

Let us write down the equations similar to (8) for the two bodies k and i, and substitute reaction forces from equations (10). Then the kinematical relation (7) will turn into the equation

$$H^T_j M^{-1}_k \left(Q_k + H_j \lambda_j - \sum_p C^T_{k_p} H_{k_p} \lambda_{k_p} \right) = H^T_j C_j M^{-1}_i \left(Q_i + H_{j_p} \lambda_{j_p} - \sum_m C^T_{i_m} H_{i_m} \lambda_{i_m} \right) + H^T_j w'_k .$$

This approach results in a system of linear algebraic equations in Lagrange multipliers λ_i for all the joints. The system has a block-three-diagonal symmetric profile for n-body chain. The profile width increases for an arbitrary constrained system but nevertheless the method remains almost linear in n. The method has a unique feature: it becomes faster when increasing the number of degrees of freedom (*DOF*) in joints because the number of Lagrange multipliers in a joint is equal to (6–*DOF*).

Several examples of n-body pendulums (with various number of *DOF* per joint) were simulated using the *Universal Mechanism (UM)* software [6]. The results show that the direct method is the fastest up to 10-15 rotational *DOF* in a chain, the composite method wins for 15 to 30 *DOF*, and further the articulated method is the best one.

3. Methods of discretization of flexible bodies. Beams and plates

3.1. FINITE RIGID SEGMENT METHOD

Following this approach, all flexible part's inertia and elastic properties are distributed among *finite rigid segments* (bodies) [2] and joints with elastic-dissipative joint forces, so the result is an ordinary rigid multibody system. The approach can be successfully applied to nonlinear cables, beams and also to plates [7], see an example in section 3.3.

The approach represents well both static and dynamic properties of the flexible part. In particular, it is effective for simulation of non-stretchable beams using relative rotational degrees of freedom due to eliminating high longitudinal parasitic frequencies.

3.2. FINITE ELEMENT METHOD

3.2.1. Floating reference frame formulation

In order to define an arbitrary 3D position of a plate element shown in Figure 3, one can use the following values: the positions r_k (denoted by r_{uv}^{00}), k=0...3, the orientations (denoted by orientation matrices A_k) and shear deformations γ_k (not shown) of the four plate edges. The orientation matrices can be specified, for example, by Cardan angles.

In terms of equations (1), the floating reference frame is associated with the edge 0. Relative displacements of the rest edges form the vector u of elastic coordinates

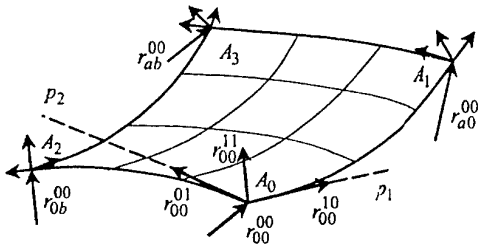


Figure 3. 3D plate finite element

$$u = \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix}, \quad u_k = \begin{Bmatrix} A_0^T (r_k - r_0) - u_k^* \\ \alpha\left(A_0^T A_k\right) \end{Bmatrix},$$

where $\alpha(A)$ is the vector function returning the Cardan angles of the rotation matrix A.

Assuming that the vector u is small, one can compute the strain and kinetic energies and derive elastic and inertia forces. However, this leads to strongly nonlinear mass matrix and generalized forces.

3.2.2. Absolute nodal coordinate formulation

Implementations of the method are known from many papers, e.g. [2,8]. Let us consider the plate element shown in Figure 3. In terms of equations (1), *no* body-fixed reference frame is used, but the nodal coordinates include all rigid-body motions of the plate.

Such a set of nodal coordinates defines an *isoparametric* plate element:

$$u = \left\{ \begin{matrix} r_{00}^{00T} & r_{00}^{01T} & r_{0b}^{00T} & r_{0b}^{01T} & r_{00}^{10T} & r_{00}^{11T} & r_{0b}^{10T} & r_{0b}^{11T} & r_{a0}^{00T} & r_{a0}^{01T} & r_{ab}^{00T} & r_{ab}^{01T} & r_{a0}^{10T} & r_{a0}^{11T} & r_{ab}^{10T} & r_{ab}^{11T} \end{matrix} \right\}^T,$$

where $r_{uv}^{ij} = \partial^{i+j} r / \partial p_1^i \partial p_2^j \Big|_{p_1=u,\ p_2=v}$ denote the vectors and their up to 2nd-order derivatives specifying the position and orientation of the four edges of the plate.

Then the position of an arbitrary point of the plate can be found as

$$r = Su = [S_{11}, S_{12}, S_{13}, S_{14}; \ldots; \ldots; S_{41}, S_{42}, S_{43}, S_{44}]u,$$

where $S_{ij} = s_i(p_1, a)\, s_j(p_2, b)\, I_{3\times 3}$ are submatrices of the *matrix of global shape functions* S depending on 1-dimensional Hermite functions

$$s_1(p,l) = s_3(l-p,l) = 1 - 3\xi^2 + 2\xi^3, \qquad s_3(p,l) = 3\xi^2 - 2\xi^3, \qquad \xi = \frac{p}{l}$$
$$s_2(p,l) = -s_4(l-p,l) = l\left(\xi - 2\xi^2 + \xi^3\right), \quad s_4(p,l) = l\left(\xi^3 - \xi^2\right),$$

used to describe the deformed state for beam finite elements [2].

Since the matrix S is u-independent, the method leads to a constant mass matrix $M = \int_0^a \int_0^b \mu S^T S \, dp_1 dp_2$ and no inertia forces. To obtain elastic forces, the following expression for strain energy of an *orthotropic plate* [9] can be employed:

$$\Pi = \Pi_{bend} + \Pi_{stretch} = \tfrac{1}{2} \int_0^a \int_0^b \left(D_1\kappa_1^2 + D_2\kappa_2^2 + (D_1\mu_2 + D_2\mu_1)\kappa_1\kappa_2 + 4D_{12}\kappa_{12}^2\right) dp_1 dp_2$$
$$+ \tfrac{6}{h^2} \int_0^a \int_0^b \left(D_1\varepsilon_1^2 + D_2\varepsilon_2^2 + (D_1\mu_2 + D_2\mu_1)\varepsilon_1\varepsilon_2 + D_{12}\varepsilon_{12}^2\right) dp_1 dp_2.$$

The preceding expression includes parameters of stiffness and sizes of the plate and its transverse and planar curvatures. The explicit expressions for the curvatures are too bulky to use them, but with the help of ideas suggested, e.g. in [8], they can be reduced to acceptable ones. Then the strain energy expression turns into "almost" quadratic form in the nodal coordinates u.

3.3. SIMULATION EXAMPLES

3.3.1. Cantilever beam subjected to large bending

This problem was simulated for comparing results obtained using both finite element formulations: the floating reference frame and the absolute nodal coordinates.

As shown in Figure 4 and in the table, the vertical force P causes large displacements of the beam free end: rotation angle θ, vertical δ_v and horizontal δ_h deflections. The rest values are the beam length L and stiffness EJ, the number of finite elements n.

Cursive values in the table correspond to the exact numerical solution of the *elastica problem* [9] and were obtained by employing the minimal amount of finite elements n.

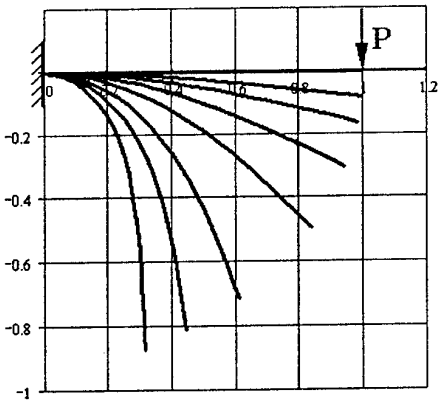


Figure 4. Cantilever beam

$\frac{PL^2}{EJ}$	Floating reference frame formulation				Absolute nodal coordinate formulation			
	n	$\frac{\theta}{\pi/2}$	$\frac{\delta_v}{L}$	$\frac{\delta_h}{L}$	n	$\frac{\theta}{\pi/2}$	$\frac{\delta_v}{L}$	$\frac{\delta_h}{L}$
0.25	3	*0.079*	*0.083*	*0.004*	2	*0.079*	*0.083*	*0.004*
	1	0.080	0.083	0.000	1	0.076	0.078	0.004
0.5	6	*0.156*	*0.162*	*0.016*	3	*0.156*	*0.162*	*0.016*
	1	0.159	0.167	0.000	1	0.143	0.138	0.012
1	7	*0.294*	*0.302*	*0.056*	4	*0.294*	*0.302*	*0.056*
	1	0.318	0.333	0.000	1	0.276	0.239	0.037
2	12	*0.498*	*0.494*	*0.160*	5	*0.498*	*0.494*	*0.161*
	2	0.523	0.521	0.132	2	0.493	0.479	0.154
5	22	*0.774*	*0.714*	*0.388*	6	*0.774*	*0.714*	*0.388*
	10	0.777	0.717	0.387	1	0.858	0.590	0.282
10	12	0.914	0.814	0.554	12	*0.911*	*0.811*	*0.555*
	6	0.921	0.822	0.551	6	0.911	0.807	0.553

3.3.2 Motion of a flexible ellipsograph with a rigid pendulum

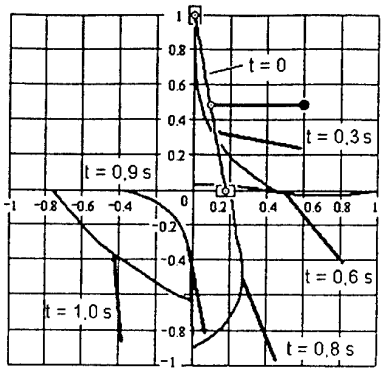


Figure 5. Flexible ellipsograph

Figure 5 demonstrates motion of a simplest rigid-elastic multibody system consisting of an elastic beam (20 finite elements, absolute nodal coordinate formulation) and a rigid pendulum attached to the middle point of the beam.

Parameters of the model are listed below:

1) beam: length $L = 1\ m$; density $\mu = 7800\ kg/m^3$; cross section $F = 10^{-4}\ m^2$ and area inertia moment $J = 10^{-8}\ m^4$; Young ratio $E = 10^8\ Pa$;

2) pendulum: length $L_1 = 0.5\ m$; mass $m_1 = 0.2\ kg$; mass inertia moment $J_1 = 0.1\ kg \cdot m^2$;

3) simulation: step $h = 10^{-4}\ s$; duration time $T = 180$ s; CPU Pentium III, 650 MHz.

3.3.3. Conveyor with hanging belt

Figure 6 presents a model of a conveyor using *UM* software [6,7] and two approaches:

1) rigid multibody system (section 3.1) having about 200 bodies, 500 degrees of freedom and 500 algebraic constraint equations;

2) finite element construction (section 3.2.1) with about 2500 nodal variables.

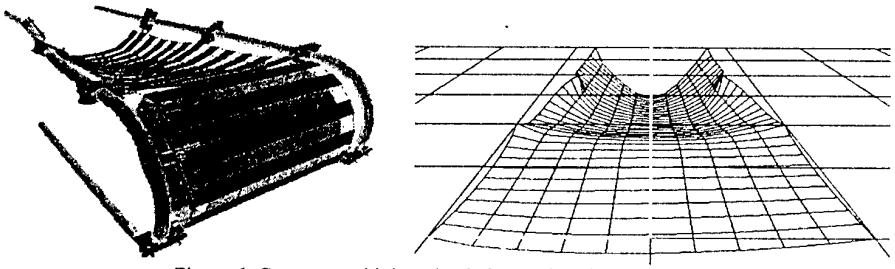


Figure 6. Conveyor with hanging belt: MBS and FEM models

4. References

1. Simeon, B. (1998) DAEs and PDEs in elastic multibody systems, *Numerical Algorithms* **19**, 235-246.
2. Shabana, A. A. (1997) Flexible multibody dynamics: review of past and recent develo pments, *Multibody system dynamics* **1**, 189-222.
3. Featherstone, R. (1987) *Robot dynamics algorithms*, Kluwer Academic Publishers, Boston.
4. Schwertassek, R. (1997) Flexible bodies in multibody systems, *Computational methods in mechanical systems* **161**, 329-363.
5. Pogorelov, D. Yu. and Dmitrotchenko, O. N. (2000) Modifications of the articulated body method for generating and solving equations of motion of multibody systems, *ICES, UNESCO International confe r-ence "Young scientists – Science, technology and professional educati on"*, 87-90 (In Russian).
6. Pogorelov, D. (1999) Multibody system approach in simulation of underwater cable dynamics, *EURO-MECH 398, Colloquium on Fluid-Structure Interaction in Ocean Engineering*, Hamburg.
7. Pogorelov, D. (1998) Plate modeling by rigid-elastic elements, *Zwischenbericht ZB-103*, Institut B für Mechanik, Universität Stuttgart.
8. Berzeri, M. and Shabana, A. A. (2000) Development of simple models for the elastic forces in the absolute nodal co-ordinate formulation, *Journal of Sound and Vibration* **235**(4), 539-565.
9. Timoshenko, S. P., and Gere J. M., (1976) *Mechanics of materials*, Mir Publishers, Moscow (In Russian).

Efficient corrector iteration for implicit time integration in multibody dynamics

Andreas Fuchs, Martin Arnold
({andreas.fuchs, martin.arnold}@dlr.de)
DLR German Aerospace Center, Institute of Aeroelasticity,
Vehicle System Dynamics Group, P.O.Box 1116, D - 82230 Wessling, Germany

Abstract. In the time integration of complex multibody system models the numerical effort is dominated by the evaluation of Jacobian matrices. The equations of motion for multibody systems result in a special structure of the Jacobian that may be exploited to save computing time. In the present paper several methods are summarized that have been made available in the industrial multibody system package SIMPACK.

Keywords: corrector iteration, time integration, evaluation of Jacobian

Abbreviations: MBS - Multibody systems; DAE - Differential algebraic equation

1. Introduction

The equations of motion of mechanical multibody systems can be derived by the principles of classical mechanics. This leads to a differential algebraic system

$$p' = v, \tag{1}$$
$$M(p)v' = \Psi(p,v) - G^T(p)\lambda, \tag{2}$$
$$0 = g(p), \tag{3}$$

where p denotes the coordinates, v the velocities, $M(p)$ the mass matrix, $\Psi(p,v)$ the vector of applied forces and momenta, $G(p)$ the constraint matrix and λ are the Lagrange multipliers.

If the equations of motion are formulated by multibody formalisms, which use relative and absolute coordinates p and q, they can be solved efficiently. The state of the MBS is completely given by p and p'. The absolute coordinates q are defined by $0 = \tilde{g}(p,q)$. The dynamical equations from the Euler-Lagrange formalism together with the second derivative of the constraint equations $0 = \tilde{g}(p,q)$ result in an index-1 system of the equations of motion

$$\begin{pmatrix} \tilde{M} & 0 & \Gamma_q^T \\ 0 & 0 & \Gamma_p^T \\ \Gamma_q & \Gamma_p & 0 \end{pmatrix} \begin{pmatrix} q'' \\ p'' \\ \lambda \end{pmatrix} = \begin{pmatrix} \tilde{\Psi} \\ 0 \\ -\xi \end{pmatrix}, \tag{4}$$

where $\Gamma_p = \partial\tilde{g}/\partial p$, $\Gamma_q = \partial\tilde{g}/\partial q$ and $\xi = \Gamma_{qt}q'' + \Gamma_{pt}p'' + \Gamma_{tt}$. The matrices Γ_p, Γ_q and $\tilde{M}$ are sparse and therefore (4) can be transformed to a banded matrix by row and column transformations. Applying a Cholesky method for banded matrices, the resulting system can be solved efficiently [3].

2. Time integration of constrained mechanical systems

The solution of a differential algebraic system is computed by an approximation y_{n+1} of the analytical exact solution $y(t_{n+1})$ at every time step. This approximation can be calculated by a modified Newton method using an approximation of the Jacobian matrix [1].

If complex mechanical systems are described by relative coordinates, the computational effort in the dynamical simulation is dominated by these evaluations of the Jacobian of the equations of motion. To reduce the computing time the special structure of this Jacobian has to be exploited in time integration.

2.1. Time integration by DASSL: basics

The integrator DASSL [1] is a special implementation of BDF methods with order and step size control. DASSL is a code for solving index zero and one systems of differential algebraic equations of the form

$$F(t, y(t), y'(t)) = 0. \tag{5}$$

In order to solve this DAE system (5) the derivative $y'(t)$ has to be replaced by a difference approximation in every time step, e.g., the first order backward difference leads to the implicit Euler formula

$$F(t_{n+1}, y_{n+1}, \frac{y_{n+1} - y_n}{h_{n+1}}) = 0, \tag{6}$$

where $h_{n+1} = t_{n+1} - t_n$. In DASSL the derivative $y'(t)$ is approximated by backward differentiation formula (BDF) of order k with $1 \leq k \leq 5$.

The solution at time t_{n+1} is calculated by a predictor-corrector method, i.e., first there is an approximation $(y_{n+1}^{(0)}, y'^{(0)}_{n+1})$ at time t_{n+1} specified and after that the final numerical solution y_{n+1} is determined by a corrector iteration.

In the corrector iteration the equation

$$F(t_{n+1}, y_{n+1}, \alpha y_{n+1} + \beta) = 0 \tag{7}$$

has to be solved with respect to y_{n+1}, whereas $\beta = y'^{(0)}_{n+1} - \alpha y^{(0)}_{n+1}$ and the parameter α is a constant, that depends on the step size and order of the used method.

The solution to (7) is evaluated by a modified Newton iteration which is given by

$$y^{(m+1)}_{n+1} = y^{(m)}_{n+1} - cJ^{-1}F(t_{n+1}, y^{(m)}_{n+1}, \alpha y^{(m)}_{n+1} + \beta) \tag{8}$$

with the Jacobian matrix J. The constant c accelerates the corrector iteration and m is a counter of the iterates. The starting value $y^{(0)}_{n+1}$ is known from the predictor as an approximation to the solution.

The required Jacobian J can be written as

$$J = \frac{\partial F(t_{n+1}, y_{n+1}, \alpha y_{n+1} + \beta)}{\partial y_{n+1}} = \alpha \frac{\partial F}{\partial y'} + \frac{\partial F}{\partial y}. \tag{9}$$

The iteration matrix J is either computed by finite differences, or supplied directly by the user.

2.2. Efficient Jacobian updates in DASSL

In consequence of the high effort needed for evaluating the Jacobian matrix, DASSL avoids reevaluations of J if possible. Often the matrices $\partial F/\partial y'$ and $\partial F/\partial y$ in (9) change very little over the span of several time steps. On the other hand, however, the parameter α changes whenever the used step size or order of the method changes.

Instead of reevaluating the iteration matrix J on every step, DASSL uses the old Jacobian as long as the derivative matrices and the parameter α have not changed very much since the last computation of J.

The Newton iteration (8) which is used by DASSL in the case of using an old Jacobian can be written as

$$y^{(m+1)}_{n+1} = y^{(m)}_{n+1} - cJ^{-1}_{old}F(t_{n+1}, y^{(m)}_{n+1}, \alpha y^{(m)}_{n+1} + \beta) \tag{10}$$

where J_{old} is the old Jacobian, saved from some previous time steps.

If the corrector iteration fails to converge then a new evaluation of the Jacobian matrix J is required [1].

3. Adapted approximation of the Jacobian in SIMPACK

The standard integrator SODASRT of the industrial MBS simulation tool SIMPACK [5] is based on the public domain solver DASSL. In order to

save computing time in SIMPACK several methods have been developed that exploit the structure of the Jacobian in MBS applications.

3.1. Difference approximation of sparse Jacobians

In SIMPACK there is implemented an adapted version of "Algorithm 618" of the Transaction on Mathematical Software (TOMS) [2]. This algorithm was originally developed for estimating sparse Jacobian matrices, if the exact sparsity structure of J is known.

In SIMPACK only an approximation of the sparsity structure is available. The exact structure could be derived from the topology of the observed model. This, however, is not yet implemented in the present SIMPACK version 8.5. Therefore, "Algorithm 618" is applied with the approximated structure.

If the sparsity structure of the Jacobian is given, "Algorithm 618" divides the columns of J into groups, such that each column belongs to one and only one group, and such that no two columns in a group have a nonzero in the same row position. A partition of the columns of J with this property is called consistent with the determination of J.

With this approach the Jacobian is not longer approximated by evaluating each column separately but all columns in one group are evaluated simultaneously by finite differences. Therefore, the approximation of the entire Jacobian matrix costs only one function call per group instead of one function call per column in the classical approach.

3.2. Partitioned evaluation of Jacobian matrices

The SIMPACK integrators offer also a partitioned evaluation method for the Jacobian J, i.e., a separate evaluation of $\partial F/\partial y'$ and $\partial F/\partial y$ in (9). This is a standard technique in the solution of ordinary differential equations, but it turned out to be very successful for DAE's too, since $\partial F/\partial y'$ has a simple structure in SIMPACK due to the implemented multibody formalisms.

The explicit formalism [5] with equations of motion

$$p' = v, \tag{11}$$

$$v' = M^{-1}(p)\Psi(p,p') - M^{-1}(p)G^T(p)\lambda, \tag{12}$$

$$0 = g(p), \tag{13}$$

leads to a Jacobian of the form

$$J = \alpha\frac{\partial F}{\partial y'} + \frac{\partial F}{\partial y} = \alpha\begin{pmatrix} I_{n_p+n_v} & 0 \\ 0 & 0 \end{pmatrix} + \frac{\partial F}{\partial y} \tag{14}$$

with the constant α that is determined by the order and step size of the integrator.

If a new evaluation J_{new} of the Jacobian matrix is necessary due to step size or order changes then the full difference approximation may be avoided whenever at least one approximation J_{old} has been computed before:

$$\begin{aligned} J_{new} &= \alpha_{new} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} + \frac{\partial F}{\partial y} \\ &\approx \alpha_{new} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} + J_{old} - \alpha_{old} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \\ &= (\alpha_{new} - \alpha_{old}) \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} + J_{old}. \end{aligned} \tag{15}$$

3.3. MBS MODELS WITH DOMINATING EXTERNAL EXCITATIONS

Recently, the partitioned evaluation was extended to approximately linear models with dominating external time excitations $u(t)$ [4].

If joint coordinates are used to set up the equations of motion, then the time excitations in rheonomic joints appear in the Jacobian matrix J. These entries of J make the time integration inefficient, because any changes of $u(t)$ result in a reevaluation of J.

In order to avoid these reevaluations the time dependent entries of J are updated whenever a partitioned evaluation is enforced, see Section 3.2.

Instead of the formulation in (5) we adapt now the standard notation of system theory and write $F = F(y, y', u(t))$. Considering the explicit formalism as before the Jacobian can be written as

$$\begin{aligned} J &= \alpha \frac{\partial F}{\partial y'}(y, y', u(t)) + \frac{\partial F}{\partial y}(y, y', u(t)) \\ &= \alpha \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} + \frac{\partial F}{\partial y}(y, y', u(t)). \end{aligned} \tag{16}$$

By Taylor expansion the new update formula is given by

$$\begin{aligned} J_{new} \approx\ & J_{old} + (\alpha_{new} - \alpha_{old}) \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} + \\ & + \sum_{i=1}^{n_u} \frac{\partial}{\partial u_i} \frac{\partial F}{\partial y}(y_0, y_0', u(t_0))(u_i(t_{new}) - u_i(t_{old})). \end{aligned} \tag{17}$$

Accordingly, updating the entries of time excitations in J requires additional calculation of the n_u partial derivatives $\partial^2 F / \partial u_i \partial y$ once at the beginning of the integration.

In SIMPACK the vector $u(t)$ contains not only the time excitations u themselves but also the first and second derivatives $u_i'(t)$ and $u_i''(t)$ for all $u_i(t)$ acting in rheonomic joints. So the additional effort consists of computing the partial derivatives $\partial F/\partial y$ with respect to u_i, u_i' and u_i''.

4. Numerical experiments

The different evaluation methods for the Jacobian are tested on various full vehicle models in SIMPACK [4]. Furthermore they were applied to a simplified benchmark problem, a chain of mathematical pendulums.

Using the sparsity structure of the Jacobian, see Section 3.1, results in a reduction of computing time up to 80% applied to the benchmark problem and up to 30% for full vehicle models in SIMPACK.

The partitioned technique of Section 3.2 yields a saving of up to 60%. Together with the methods of Section 3.1 the cpu-time may be reduced even more.

With the new partitioned algorithm of Section 3.3 the cpu-time is reduced by up to 90% for a chain of pendulums. As part of an industrial project the algorithm has been applied as well to a complex automotive model in SIMPACK resulting in savings up to 40%.

References

1. K.E. Brenan, S.L. Campbell, and L.R. Petzold. *Numerical solution of initial-value problems in differential-algebraic equations.* SIAM, Philadelphia, 2nd edition, 1996.
2. T.F. Coleman, B.S. Garbow, and J.J. Moré. Algorithm 618: FORTRAN subroutines for estimating sparse Jacobian matrices. *ACM Transactions on Mathematical Software*, 10:346–347, 1984.
3. E. Eich-Soellner and C. Führer. *Numerical Methods in Multibody Dynamics.* Teubner–Verlag, Stuttgart, 1998.
4. A. Fuchs. Effiziente Korrektoriteration für implizite Zeitintegrationsverfahren in der Mehrkörperdynamik. Master's thesis, Munich University of Technology, Department of Mathematics, 2002.
5. W. Rulka. *Effiziente Simulation der Dynamik mechatronischer Systeme für industrielle Anwendungen.* PhD thesis, Vienna University of Technology, Department of Mechanical Engineering, 1998.

QUASI-STATIC MOTION OF THE TWO-LINK AND THREE-LINK MECHANISMS ALONG A HORIZONTAL PLANE

T.YU.FIGURINA
Researcher, Institute for Problems in Mechanics of the Russian Academy of Sciences 101-1 Prosp. Vernadskogo, Moscow 119526, Russia

Abstract.

We investigate the possibility of slow (quasi-static) locomotion of multi-link systems along a horizontal plane owing to changing their configurations. It has been shown in [1] that by alternating slow and fast phases of motion two-link system and three-link system with links connected in series can move along itself, sideways, and rotate on a spot. We prove [2] that the quasi-static motion of a two-link system is uncontrollable and that the trajectories of the system's vertices are uniquely defined by the initial position of the system. We show that there exist much enough possibilities for the quasi-static motion of a three-link system with the links connected in star. One can arrange a slow motion with the central vertex of the system moving along a prescribed line on the plane.

1. Statement of the problem

We consider a two-link and a three-link systems, both lying on a horizontal rough plane with dry friction force. The control torques are applied at the system's joints and directed perpendicularly to the plane. We consider quasi-static motions of the systems, i.e., the motions with infinitesimal velocity and acceleration. The problem is to investigate the possibilities of slow locomotion of these systems and to find out how to drive the systems to an arbitrary position on the plane, if possible.

Three-link system $A_0A_1A_2A_3$ consists of three identical weightless links A_0A_i, $i = 1, 2, 3$, connected in star by a joint with two motors generating the control torques M_{i3} acting between the links A_0A_3 and A_0A_i, $i = 1, 2$ (Figure 1). Three point masses, each being equal to m, are located at the

free ends of the system, and the mass equal to m_0 is located at the vertex A_0. Denote by μ the ratio of the masses, $\mu = \frac{m_0}{m}$, and by l the length of each link A_0A_i.

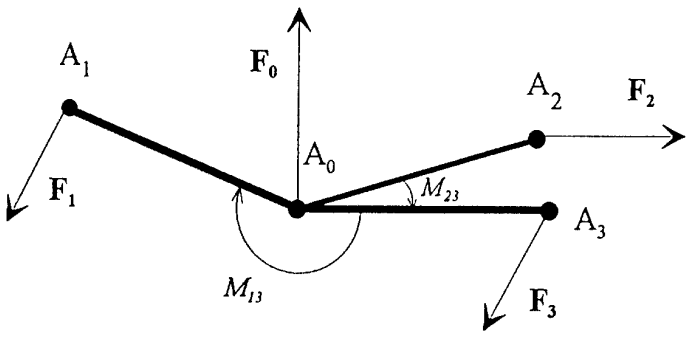


Figure 1. Three-link system

Two-link system $A_1A_0A_2$ consists of two weightless links connected by a joint with a motor. Let m_i be the mass located at the vertex A_i, $i = 0, 1, 2$ and let l_j be the length of the link A_0A_j, $j = 1, 2$. To avoid an ambiguity we suppose that $m_1l_1 \neq m_2l_2$.

We will use the unified notation and equations of motion applying both to the two-link and to the three-link systems. Let $\mathbf{F}_i$ be dry friction force acting at the vertex A_i, M_i the moment of this force, and $\mathbf{v}_i$ and v_i the velocity vector of the vertex A_i and its modulus, $i = 0, 1, 2, (3)$. Dry friction force obeys Coulomb's law written in an appropriate system of units as

$$\mathbf{F}_i = \begin{cases} -\mathbf{v}_i/v_i, & v_i \neq 0 \\ \forall \mathbf{e}, |\mathbf{e}| \leq 1, & v_i = 0 \end{cases} \quad i = 0, 1, 2, (3) \tag{1}$$

A necessary equilibrium condition for each of the multi-link system is expressed as follows:

$$\sum_0^{2(3)} \mathbf{F}_i = 0 \tag{2}$$

$$\sum_0^{2(3)} M_i = 0 \tag{3}$$

If this condition is satisfied, then we can uniquely choose the control torque for the two-link system (the control torques M_{13} and M_{23} for the three-link system) such that the equilibrium condition for each link of the system is also satisfied. The velocities of the vertices are subjected to the following constraints:

$$(\mathbf{v}_i - \mathbf{v}_0, \overrightarrow{A_0A_i}) = 0, \quad i = 1, 2, 3 \tag{4}$$

The system of equations (1) – (4) governs the quasi-static motion of the two-link (three-link) system. The solution of (1) – (4) is a triple (a four) of vectors $(\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, (\mathbf{v}_3))$.

2. Slow Motions of the Two-Link System

Let γ be the angle between the vectors $\overrightarrow{A_1A_0}$ and $\overrightarrow{A_0A_2}$, $\gamma \in [-\pi, \pi]$. The projections of the velocities $\mathbf{v}_i$ onto the axes of a coordinate system attached to the link A_0A_1 depend only on the angle γ.

Proposition 1. The solution of the quasi-static motion problem (1)-(4) is uniquely defined by the position of the two-link system.

The trajectories of the system's vertices are uniquely defined by the initial position of the system. Let I-, II-, and III-motions be the motions of the two-link system with 1,2 and 3 moving vertices. Let I^k and II^k be I- and II-motions with the vertex A_k moving, $k = 1, 2$ (when II^k-motion occurs, the vertices A_0 and A_k are moving). Denote by γ_{ij}^k the angle separating I^k-motion and J^k-motion, i.e., such an angle that if $\gamma < \gamma_{ij}^k$ then I^k-motion occurs and if $\gamma > \gamma_{ij}^k$ then J^k-motion occurs (quasi-static I^i- and II^j-motions with $i \neq j$ never follow each other).

We omit here the description of the full solution of (1)-(4) and the expressions for γ_{ij}^k. We will outline some featuers of the quasi-static motion. When quasi-static II-motion occurs, one end vertex A_i moves along a straight line such that the distance between the line and another end vertex A_j is equal to $\frac{m_j l_j}{m_i}$. When quasi-static III-motion occurs, the vertices A_i move along the straight lines which intersect at one point. The trajectory of each vertex consists of the smoothly joining straight line segments and the arcs of the circles. The sequence of the motion phases depends on the parameters of the system. It may look, for example, as $I^1 \to II^1$ or as $II^1 \to III \to I^2 \to II^2 \to III \to II^1$ for γ increasing from 0 to π.

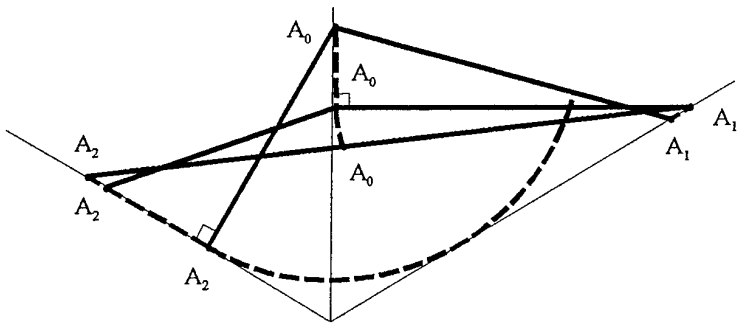


Figure 2. Two-link system with equal masses

As an example, consider the two-link system with equal masses $m_1 = m_0 = m_2$. Let $l_2 < l_1/2$. Then II^2-motion occurs for $\gamma \in [0, \gamma_{23}^2]$, III-motion occurs for $\gamma \in [\gamma_{23}^2, \gamma_{31}^2]$, and I^2-motion occurs for $\gamma \in [\gamma_{31}^2, \pi]$. The trajectories of the vertices for $\gamma \in [0, \pi]$ and the two-link system positions corresponding to $\gamma = 0, \gamma_{23}^2, \gamma_{31}^2, \pi$ are shown in Figure 2.

3. Slow Motions of the Three-Link System

Consider first slow motions of the three-link system with fixed vertex A_0. We will not try to describe all such motions but will find only some motions, depending on the parameter μ.

Proposition 2.

(i) Let in both the initial and the terminal states of the three-link system the position of the link A_0A_1 be the same and let the other two links be close enough to each other, i.e., $\angle A_2A_0A_3 \leq 2 \arcsin \frac{\mu}{2}$. Then we can quasi-statically drive the system from the initial state to the terminal one, with the link A_0A_1 being fixed and the other two links rotating in turns remaining close enough to each other.

(ii) Let in both the initial and the terminal states of the three-link system the position of the vertex A_0 be the same and let all three links are close enough, i.e., for $\forall i \; \exists j \neq i$ such that $\angle A_iA_0A_j \leq 2 \arcsin \frac{\mu}{2}$. Then we can quasi-statically drive the system from the initial state to the terminal state. At each time instant, one of the links rotates and the other two links are fixed, all three links being close enough to each other.

(iii) Let $\mu \geq \sqrt{3}$. Then we can quasi-statically drive the three-link system from an initial state to an arbitrary terminal state with the same position of the vertex A_0, with two link being fixed and one link rotating in turns.

Consider now motions of the three-link system with moving central vertex.

Proposition 3.

If $\mu \geq 2$, then the quasi-static motion of the three-link system with moving central vertex is impossible. If $\mu < 2$, then there exists an infinite set of solutions of (1)-(4).

3.1. EXAMPLE : LOCOMOTION WITH $\mu = \sqrt{3}$ AND $\mu < 1$

Consider the three-link system with $\mu = \sqrt{3}$ and describe its motion with the vertex A_0 moving along a straight line l_0, $A_0 \in l_0$. First, turn the links so that $A_0A_2 \perp l_0$, $\angle A_2A_0A_3 = \frac{5\pi}{6}$, and the distance from the point A_1 to the straight line A_0A_2 is equal to $1 - \frac{\sqrt{3}}{2}$ (see Figure 3). Denote by l_i the straight lines, $A_i \in l_i$, $l_1 \perp l_0$, $l_3 \parallel l_2$, $\angle l_0l_2 = \frac{\pi}{6}$ and let $\mathbf{v}_i \in l_i$. Then

the equilibrium conditions of (1),(2) are fulfilled, and the vertices can move along the lines l_i while the constraint of (4) is satisfied. The motion comes to a complete stop when $A_0A_1 \perp l_0$, the vertices A_2, A_0, A_3 lie on a straight line and $\angle A_1A_0A_2 = \frac{\pi}{12}$. To shift the vertex A_0 along the line l_0 again, one should turn the links to bring them to the positions parallel to those before the first shift. At any time instant, one can stop the motion and change the direction of $\mathbf{v}_0$. Hence, the central vertex can move along a prescribed broken line on the plane.

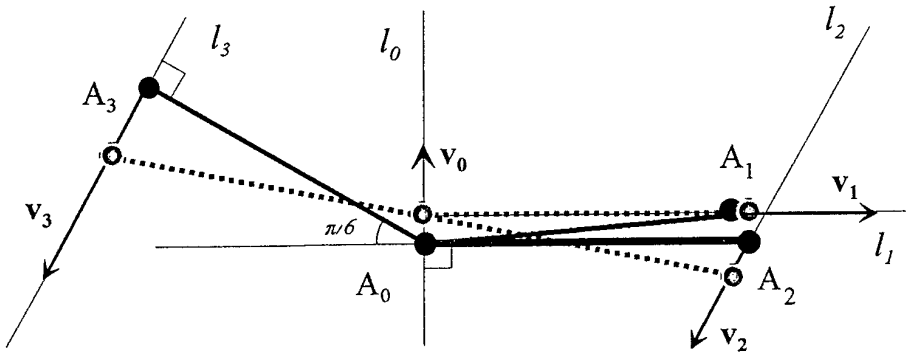


Figure 3. Locomotion along a straight line, $\mu = \sqrt{3}$

Consider now the three-link system with $\mu < 1$. Let at the starting instant the positions of all three links be the same and let us want to shift the vertex A_0 along a straight line l_0.

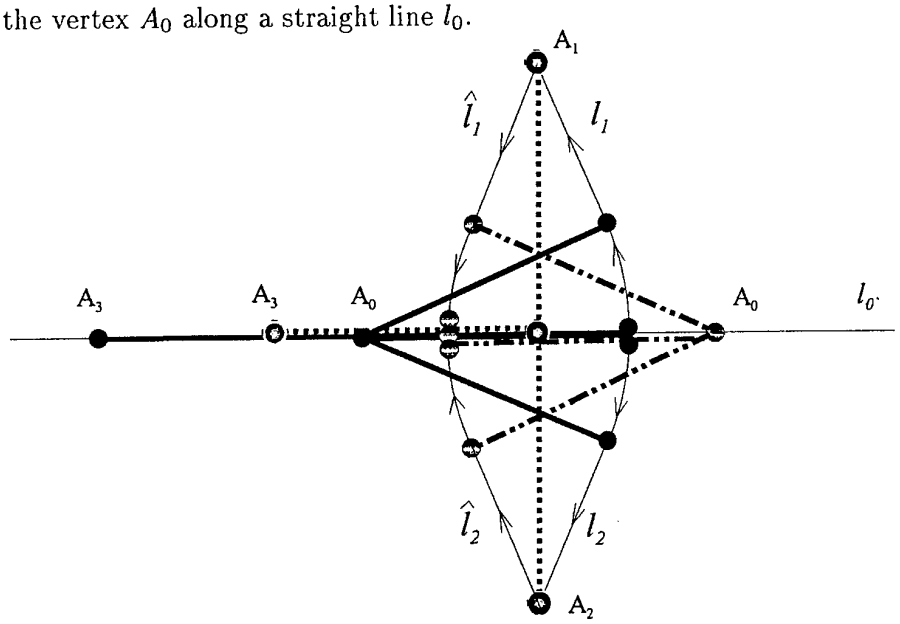


Figure 4. Locomotion along a straight line, $\mu < 1$

First, we turn the links so that all of them lie on l_0 and coincide with each other. Then we fix the link A_0A_3 and turn both links A_0A_1 and A_0A_2 to the same position which is opposite to the link A_0A_3. Then we begin to move aside the links A_0A_1 and A_0A_2 symmetrically with respect to the axis l_0. While $\angle A_1A_0A_2 < 2\arcsin\frac{1+\mu}{2}$, the link A_0A_3 remains fixed. When $\angle A_1A_0A_2 \in [2\arcsin\frac{1+\mu}{2}, \pi]$ the vertices A_i move along the straight lines l_i and $\hat{l}_i$, $i = 1, 2$ and the vertices A_0 and A_3 move along l_0. At the end of the motion, the positions of all links are the same again and the motion direction may be changed. Using this motion algorithm, one can drive the three-link system to any prescribed state on the plane, but the trajectory of the central vertex cannot be chosen arbitrarily.

4. Acknowledgments

This work was carried out under financial support of the Russian Foundation for Basic Research (grant No. 02-01-00201) and the Grant for Support of Leading Scientific Schools (00-15-96013).

5. References

1. F.L.Chernousko (2000) The motion of a multilink system along the horizontal plane, *J.Appl. Math. Mech.*, Vol..64, No.1, 5-15

2. T.Yu.Figurina (2001) Quasi-static motion of two-link system along the horizontal plane, *Proc. Fourth Int. Conf. on Climbing and Walking Robots*, 497-504.

KINEMATIC ANALYSIS OF MECHANISMS IN THE NEIGHBOURHOOD OF SINGULAR POSITIONS USING GENERAL NUMERICAL CONTINUATION METHODS.

J. FRĄCZEK
Warsaw University of Technology
Nowowiejska 24, 00-665 Warsaw, POLAND

1. Introduction

The algorithm of kinematical analysis in absolute coordinates is usually based on the trajectory tracing using time as the independent parameter and classical Newton iteration scheme [2]. In case of analysis of complicated mechanism, simulation usually fails in singular positions and the reason of that cannot be easily detected by the user. Simultaneously singular positions analysis can provide interesting information from the mechanism synthesis and control system synthesis point of view.
The paper presents an algorithm of numerical continuation using local parameterization instead of time parameterization. The simplest cases of singularities like simple bifurcation points or turning points can be detected and analyzed. The main idea is to analyze firstly mechanism using time or local parameterization with test function being responsible for singularity detection. The trajectory can be investigated in detail in the intervals suspected for bifurcation or turning points detected by test function during introductory simulation.
The example of such an algorithm was implemented by author in his test computer program.

2. Classical algorithm of kinematical analysis

The collection of all constraints induced by the joints present in multibody model is denoted by [2], [3]:

$$\mathbf{\Phi}^k(\mathbf{q},t)=\mathbf{0} \tag{1}$$

and $\mathbf{q}$ is the vector of absolute generalised coordinates [3].
The number of scalar equations in (1) is equal to l and the number of generalized coordinates is equal to $N=6m$ where m is the number of rigid bodies (three variables for rotation parameterization are used). Typically, $l<N$.
A motion is represented as a time dependent constraint equation *(driving constraints)*:

$$\Phi^d(\mathbf{q},t)=\mathbf{0} \tag{2}$$

Revisiting the definition of the position, velocity and acceleration kinematic constraint equations, for constraint equations induced by either joints or motions in the most general case the following equations must be satisfied at any time t (*position level analysis*):

$$\Phi(\mathbf{q},t)=\begin{bmatrix}\Phi^k(\mathbf{q},t)\\ \Phi^d(\mathbf{q},t)\end{bmatrix}=\mathbf{0} \tag{3}$$

and (*velocity and respectively acceleration level analysis*)

$$\frac{d\Phi(\mathbf{q},t)}{dt}=\dot{\Phi}(\mathbf{q},t)=\Phi_{\mathbf{q}}\dot{\mathbf{q}}+\Phi_t=\mathbf{0} \tag{4}$$

$$\frac{d^2\Phi}{dt^2}=\Phi_{\mathbf{q}}\ddot{\mathbf{q}}+(\Phi_{\mathbf{q}}\dot{\mathbf{q}})_{\mathbf{q}}\dot{\mathbf{q}}+2\Phi_{t\mathbf{q}}\dot{\mathbf{q}}+\Phi_{tt}=\Phi_{\mathbf{q}}\ddot{\mathbf{q}}-\boldsymbol{\Gamma}=\mathbf{0} \tag{5}$$

If constraints (3) are independent in the point $\breve{\mathbf{q}}_0=[\mathbf{q}_0^T,t_0]^T$ (regular point) i.e.:

$$\text{rank}\,(\Phi_{\mathbf{q}})=N \tag{6}$$

then exist unique solutions of linear systems (4) and (5) and nonlinear system (3) in the neighbourhood of the point $\breve{\mathbf{q}}_0=[\mathbf{q}_0^T,t_0]^T$. From numerical point of view the kinematical analysis of the system described by (3) can be considered as a numerical tracing of a trajectory T such that:

$$T=\left\{\breve{\mathbf{q}}=[\mathbf{q}^T,t]^T:\Phi(\breve{\mathbf{q}})=\mathbf{0},\ \mathbf{q}=\mathbf{q}(t),\quad t_0<t<t_1,\ \breve{\mathbf{q}}\in R^{N+1}\right\} \tag{7}$$

In general, numerical tracing of the trajectories (7) is the subject of *numerical continuation methods* [1]. One of the simplest methods is Euler predictor-Newton corrector scheme. Classical Newton corrector can be defined in the form:

$$\Phi_{\mathbf{q}}(\mathbf{q}^k,t^{i+1})\Delta\mathbf{q}^k+\Phi(\mathbf{q}^k,t^{i+1})=\mathbf{0} \tag{8}$$

The iterative algorithm (8) is numerically very efficient under strong assumption that condition (6) is fulfilled (i.e. all points of the trajectory are regular) and if good starting positions is chosen. In case the trajectory contain singular points, equation (8) becomes ill-conditioned and simulation usually fails.

We will discuss very simple cases of singular configurations and propose algorithm for numerical detection of such points. For simplicity it is assumed that redundant constraints are eliminated from the system (3).

3. Singular positions – examples. Multilink robot analysis and synthesis

Practical example of singular behavior is multilink robot structural synthesis and analysis. The multilink robot designed in IAAM consists of several sets of bodies called segments. The robot was primarily intended to weld car body in places, which are not easy to reach. Each segment is built of rigid parts and its kinematical scheme is given in the Fig. 1. The segment consists of n rigid parts connected by spherical-translational (II-th class) and revolute joints. The mobility of the segment (Grubler number) does not depend on number of rigid bodies and is always equal to 2. The segment does not contain redundant constraints. The 2,3 4 (generally m) segments with different or equal number of bodies can be connected giving manipulator with Grubler count equal to 4,6 and 8 ($2m$) respectively.
The kinematical structure of the robot is investigated for different trajectories of the robot tip. The initial position of the robot is known.

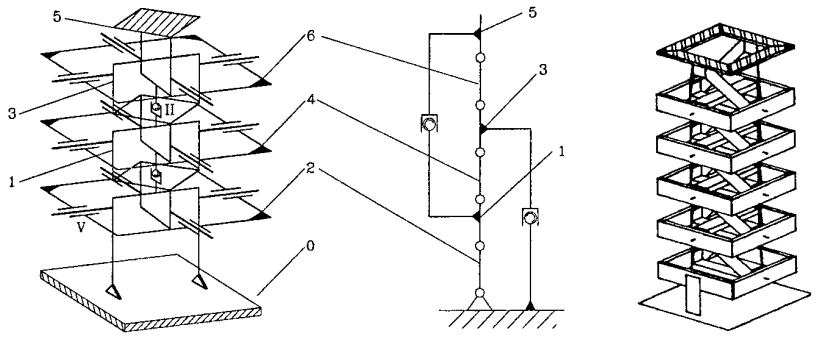


Figure 1. The kinematical scheme of one segment of the multilink robot. The mobility (Grubler number) is equal to 2 and does not depend on number of rigid bodies.

It is assumed that multilink robot is built of three segments (6 DOF total) with different number of bodies in each of the segment - equal to 6,8,10,12 and 14 respectively. Robot is driven in revolute joints – two driving torques in the lowest revolute joints (at the base of the segment) for each segment. The relative positions of kinematical pairs of the segments are parameterized in certain range. For kinematical and structural synthesis the one of the trajectories obtained from technical specifications of the welding process was chosen.
The kinematical and structural synthesis consists of three tasks: checking whether trajectory is accessible for given structure and dimensions of the robot (detection of lock-up positions), detection of singular positions of the robot and particularly positions where kinematical parameters of robot links are not continuous function of time and determining the relative angles in the selected revolute joints (with actuators) as the function of time (inverse kinematics).
Two kinds of singular configurations are shown in the Fig. 2 and Fig. 3. Figure 2 presents results of simulation of kinematical analysis for robot consisting 6 bodies in

each segment carried out with commercial multibody package (ADAMS). The planned trajectory is not accessible for that structure of robot due to the lock-up configuration detected for time of simulation equal approximately to 1.5s.

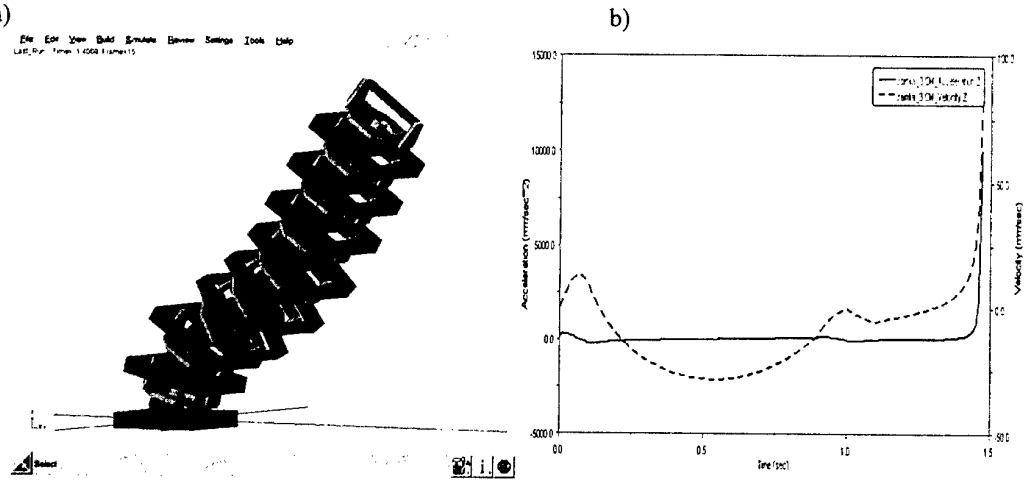


Figure 2. The results of simulation for robot with 3 segments consisting 6 bodies. Lock-up detection: a) robot position in lock-up configuration, b) diagrams of velocities and acceleration of cm marker of one of the robot body (3rd segment).

In the Figure 3 the results of simulation of robot with 8 bodies in each segment are shown. The diagram of velocities (Fig. 3c) indicates that in time of simulation equal to 2.54s solution of the kinematical task switches discontinuously from one branch of the solution to the another. Figures 3a and 3b show two consecutive positions of the robot confirming discontinuity of the solution obtained from simulation with multibody package.

4. Branch tracing using numerical continuation methods

The Newton corrector (8) can be replaced with more general corrector given by:

$$\breve{\mathbf{q}}^{k+1} = \breve{\mathbf{q}}^{k} - \Phi_{\breve{\mathbf{q}}}^{+}(\breve{\mathbf{q}}^{k})\Phi(\breve{\mathbf{q}}^{k}) \quad \text{i} \quad \breve{\mathbf{q}}^{0} = \widehat{\breve{\mathbf{q}}}^{i+1} \tag{9}$$

where $(.)^{+}$ denotes pseudo-inverse matrix (Moore-Penrose) [1].

Pseudo-inverse matrix can be calculated efficiently using inverse matrix to matrix **J** given by the formula [1]:

$$\mathbf{J}^{-1} = \begin{bmatrix} \Phi_{\mathbf{q}} & \Phi_{t} \\ \mathbf{e}^{T} & \varsigma \end{bmatrix}^{-1} = \begin{bmatrix} \Phi_{\mathbf{q}}^{-1} + \Phi_{\mathbf{q}}^{-1}\Phi_{t}s^{-1}\mathbf{e}^{T}\Phi_{\mathbf{q}}^{-1} & -\Phi_{\mathbf{q}}^{-1}\Phi_{t}s^{-1} \\ -s^{-1}\mathbf{e}^{T}\Phi_{\mathbf{q}}^{-1} & s^{-1} \end{bmatrix}, s = \varsigma - \mathbf{e}^{T}\Phi_{\mathbf{q}}^{-1}\Phi_{t} \tag{10}$$

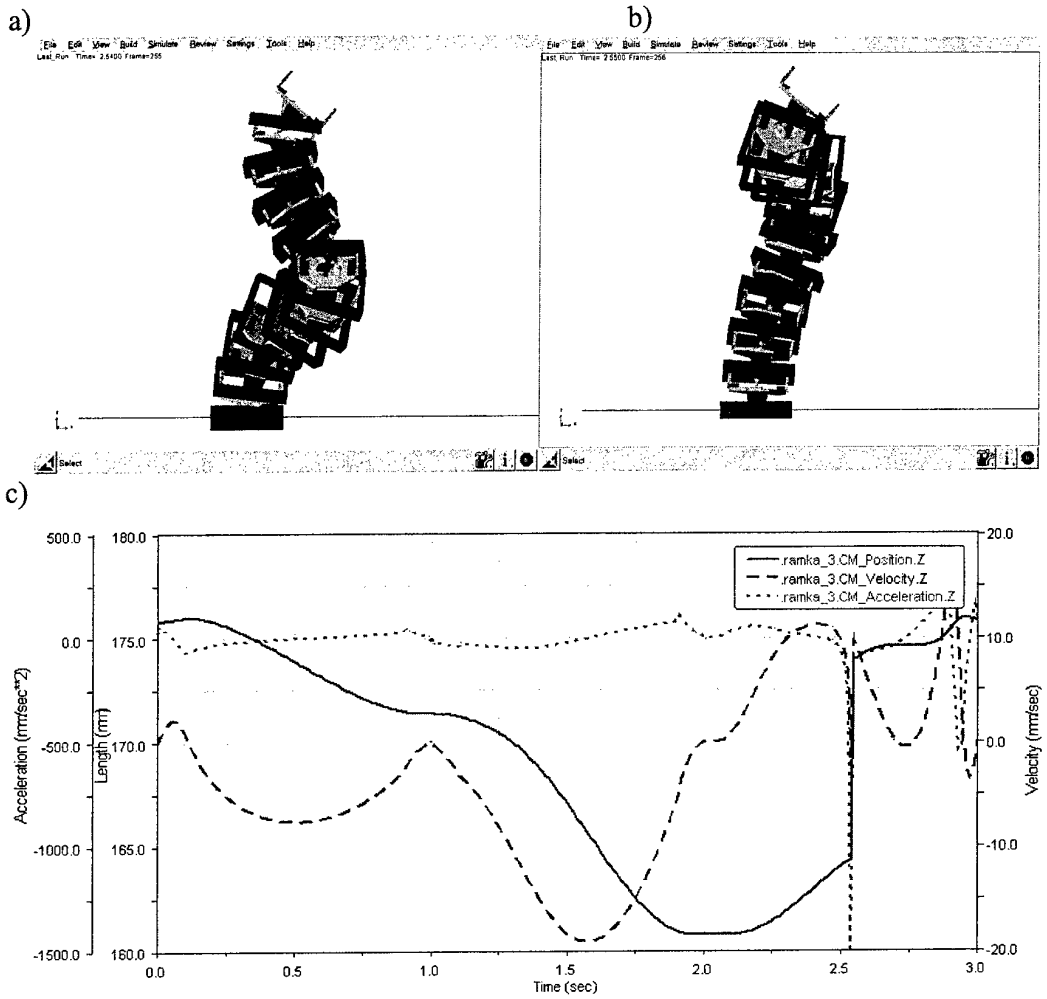


Figure 3. The results of simulation of multilink with 3 segments consisting 8 bodies. At time=2.54 s singular position and switch of the solution from one branch to the another can be observed.
a),b) The animation sequences of robot in time=2.54s and time=2.55s. c) diagrams of position, velocity and acceleration of cm marker (z axis) on one of the robot body (in 3 rd segment).

In order to trace trajectory (branch) numerically, local parameterization strategy was chosen. Instead of the time parameter other parameter is chosen as independent. For parameter choice the parameter ζ in the formula (10) is responsible. The independent parameters are constant in the intervals of time. Moore-Penrose matrix can be calculated efficiently with the formula:

$$\dot{\mathbf{\Phi}}^{+}_{\dot{\mathbf{q}}} = (\mathbf{I} - \mathbf{s}\mathbf{s}^{T})(\mathbf{J}^{-1})_{N} \tag{11}$$

where $\mathbf{s}$ is tangent vector and $(\mathbf{J}^{-1})_{N}$ denotes submatrix of matrix (10) consisting of the first N column.

It should be pointed out that, in the algorithm of matrix $\mathbf{J}^{-1}$ evaluation, sparsity of the matrices can be easy exploited. To detect singular configuration we define two simplest cases of singularities – turning point and simple bifurcation point [1].

Point $\breve{\mathbf{q}}_0 = [\mathbf{q}_0^T, t_0]^T$ such that $\mathbf{\Phi}(\breve{\mathbf{q}}_0) = \mathbf{0}$ is a:

1. *Turning point* (*limit point* or *fold bifurcation*) if rank $(\mathbf{\Phi}_{\mathbf{q}}) = N\text{-}1$ and rank $(\mathbf{\Phi}_{\breve{\mathbf{q}}}) = N$ and there is a parameterisation $\mathbf{q}(\tau)$, $t(\tau)$ with $\mathbf{q}(\tau_0)=\mathbf{q}_0$ and $t(\tau_0)=t_0$, and $d^2t/d\tau^2 \neq 0$.
2. *Simple stationary bifurcation point* if rank $(\mathbf{\Phi}_{\mathbf{q}})$ = rank $(\mathbf{\Phi}_{\breve{\mathbf{q}}}) = N\text{-}1$ and exactly two branches of solutions intersect with two distinct tangents.

If during branch tracing turning point is detected, Jacobian matrix $\mathbf{\Phi}_{\mathbf{q}}$ becomes singular, but numerical scheme defined by (10) and (11) can still be applied on condition that time is not independent parameter.

For bifurcation point detection the test function ζ is introduced which is evaluated during the branch tracing. A bifurcation is indicated by a zero of ζ – that is a branching test function satisfies the property $\zeta=0$ [4]. For the branch tracing given by iterative scheme (10) the test function can be proposed in the form $\zeta = \det\begin{pmatrix} \mathbf{\Phi}_{\breve{\mathbf{q}}} \\ \widetilde{\mathbf{J}}^T \end{pmatrix}$. This expression can be evaluated very efficiently taking into consideration sparsity of the Jacobian matrix $\mathbf{\Phi}_{\mathbf{q}}$.

In the close neighborhood of the bifurcation point singular position can be evaluated with grater accuracy using direct or indirect method for calculating branch [4]. In direct method the equation set can be extended to the new branch system [4]:

$$\widetilde{\mathbf{\Phi}}(\mathbf{Y}) = \begin{bmatrix} \mathbf{\Phi}(\mathbf{q},t) \\ \mathbf{\Phi}_{\mathbf{q}}(\mathbf{q},t)\mathbf{h} \\ h_k - 1 \end{bmatrix} = \mathbf{0}, \quad \mathbf{Y} = [\mathbf{q}^T, t, \mathbf{h}^T]^T \tag{12}$$

where $\mathbf{h}$ is a tangent vector.

System (12) can be solved using efficient numerical solvers for sparse matrix equation and Newton-like iterative scheme.

It should be pointed out that presented algorithms can be also used for branch switching i.e. calculating one (at least) solution on the emanating branch. If one of one solution is situated somewhat close to the bifurcation point then other solution on the emanating branch can be found using techniques of perturbations widely used in numerical continuation theory [1].

Using general techniques for branch tracing and switching described above the general numerical test program was developed. Its idea was based on the research package named BIFPACK [4].

5. An example of singular point detection.

The model of multilink robot was analysed this time in the author computer test programme. Kinematics of the robot was described in the absolute coordinates and joint constraints. In the Fig. 4 one of the simulation results is shown. The turning point is detected, which is geometrical interpretation of the lock-up position of the mechanism.

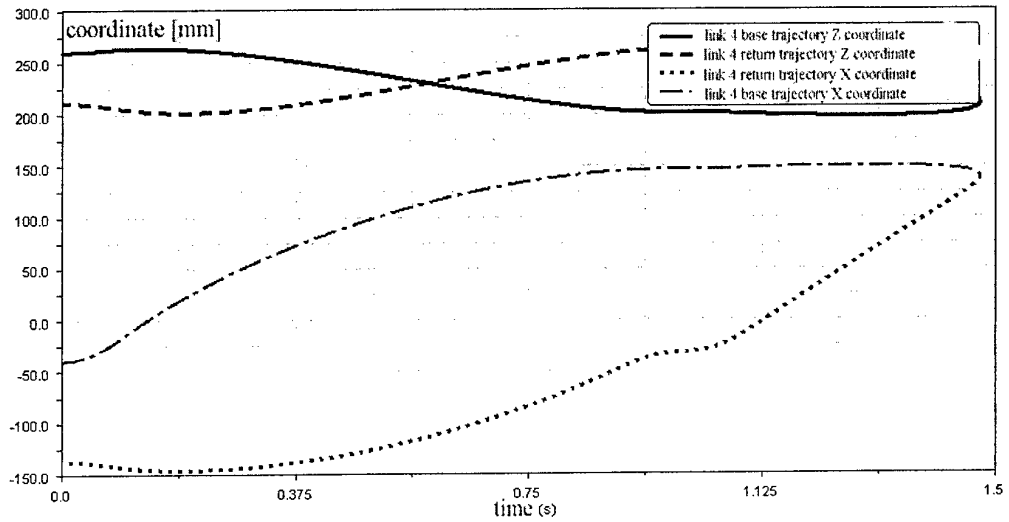


Figure 4. Global coordinates (x, z) of the last link centre of multilink robot.

6. Conclusion.

The algorithm based on local parameterisation is proposed in the paper. Two cases of singular points like turning point and bifurcation point can be detected with this algorithm. Moreover branch switching can be easily implemented. Basing on the presented formulas computer test program is built which can be prototype for general multibody module intended to detailed analysis of simple singular points. Detection of singular position is extremely important e.g. in robotics where control synthesis or actuator synthesis requires this type of information.

7. References.

1. Allgower E.L., Kurt G. (1990) *Numerical Continuation Methods*, Berlin Heildelberg, Springer Series in Mathematics.

2. Garcia de Jalon, Eduardo Bayo (1994) *Kinematic and Dynamic Simulation of Multibody Systems*, Springer-Verlag.
3. Haug E.J (1989) *Computer-Aided Kinematics and Dynamics of Mechanical Systems, Volume I: Basic Methods*, Allyn and Bacon.
4. Seydel R. (1997) Nonlinear Computation. *Int. Journal of Bifurcation and Chaos*, vol.7, 2105-2126.

MECHATRONIC APPROACH FOR SIMULATION OF ROBOTS AND WALKING MACHINES

Krassimir E.Georgiev – Assoc. Prof. Dr. Eng. , IMEH BAS
Teodora Ivanova

ABSTRACT

This paper present a new mechatronic approach to simulation of complex multibody systems, based on virtual modelling of manipulation robots and walking machines. Some basic examples concerning to robotized assembly systems and devices are described.

1.INTRODUCTION

The typical mechatronic system has at least some of the following features:[1,2,3]

- Immediate effect on the environment by controlled motion or application of the technological forces ,
- Level of adaptivity, flexibility, reprogrammability and inteligence,
- Modular structure, built on the basis of optimization criteria for low energy consumption, high speed motions and high accuracy of positioning
- In-built modules(mechanical, sensor, driving, control) as a system of the whole structure..

Multysensor robot systems and robotized assembly structures are typical objects of mechatronics and are characterized by a great complexity and an improved functionality, which is achieved by an integration of mechanical, electrical and electronic functional elements. All avaiable physical principles are to be considered for an optimal solution. Mechatronic systems achieved their performance by integration on two leveles:

- Integration of the optimised components, using information processing .The design of such systems on the level of transfer and state functions is the task of system theory
- Integration of the functional components into the structure of the mechatronic system, which is optimised as a whole ,including the capabilities of information processing [4,5,6,7,8].

2. HIERARCHICAL APPROACH TO INTEGRATION OF MECHATRONICAL SYSTEM

In a complex robotic system the essence of the multisensor (actuator} integration is data reduction and data fusion. The creation of a mechatronic

system can be viewed as a network of interconnected processing sub-systems, by using of various sensors,actuators and processors. From the processing subsystem viewpoint the integration consists of:
-Hardware design of sensors,actuators and processors,
-Signal and image processing,
-Dynamic control,
-Mathematical modelling,
-Development tools for subsystem design,
■ Real time operation.
The following tasks must be solved as algorithmic base of the suggested approach :
■ knowledge representation of the information and processing ,
■ system satisfaction of real-time constraints imposed by the environment - virtual architecture of the mechanical and processing subsystems,
■ physical architecture of the actuators, sensors and processors,
■ software (hardware) tools for virtual simulation and optimization

3. MECHATRONIC UNIT FOR ADAPTIVE ASSEMBLY

3.1. Integrated design of structure and control

MATLAB software package provides powerfull tools for design of mechatronic systems. Here we suggest an integrated approach to synthesis by applying of kinematic and dynamic components (offline knowledge base), connected in virtual world, which is displayed on the graphic components. Then we create "virtual robots" that simulate the robot, s behavior and manufacturing cell as well (cooperating robots in the tasks of manufacturing). This process is performed in a sequantical way on a power computer with graphic vizualization (Fig.1).

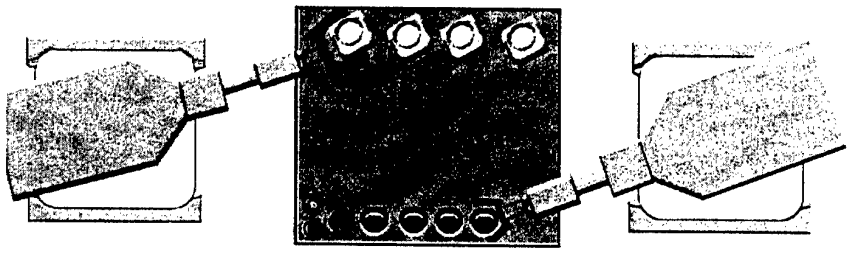


Figure 1. Virtual simulation of mechatronic structures

After off-line simulation of the closed loop, s block diagrams in SIMULINK the verification of new control strategies with the real plant can be done in a few minutes.

3.2. **Structure of the mechatronic unit for adaptive assembly**

Mechatronic unit for adaptive assembly, consists of mechanical, sensing and control subsystems [4,5,6]. The mechanical system is designed by two robots - main and assisting and basic station. The main robot is of PUMA type, built from the basic type components and to increase the adaptive possibilities of the robot, a mechatronic adaptive device is applied [Georgiev,1995]. The main robot (Mr) is designed to convey the components of the assembled parts and to perform the assembly.(Fig.2)

The assisting robot (Ar) is of SCARA type, designed to provide for fast transportation and positioning of the based assembly parts. This robot is characterized with two zones of the action - technological (for direct assembly) and auxiliary (for feeding manipulations).

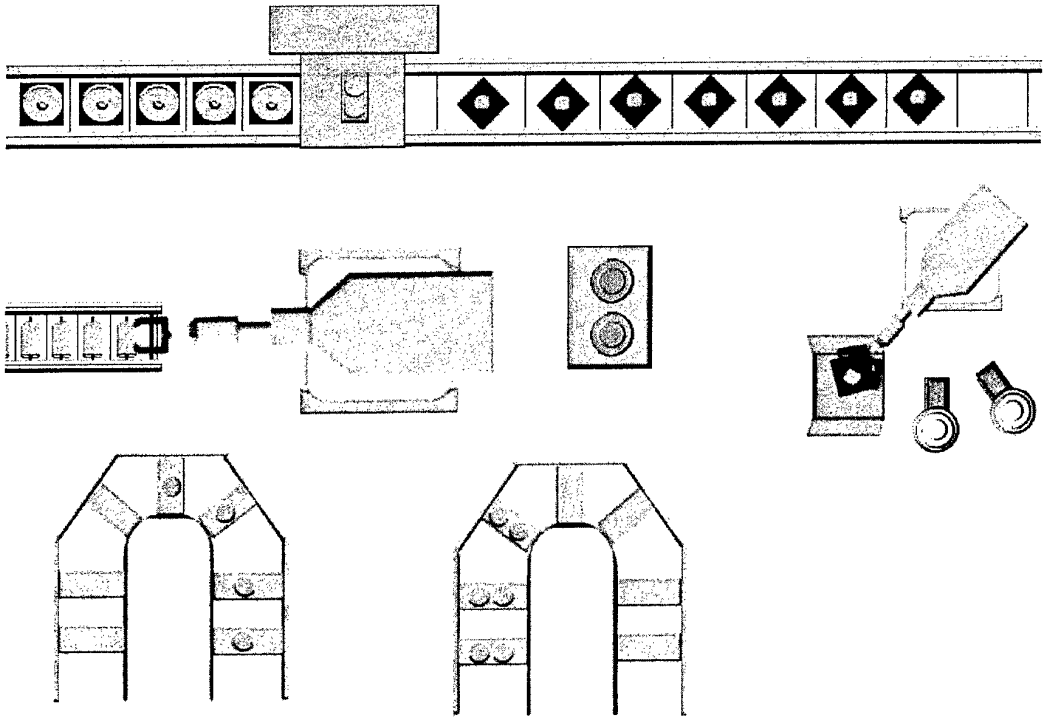


Figure 2. A structure of mechatronic unit

The basic station (Bs) is coupled with six component force-torque sensor and is applying for precize measurements of the contact forces and fine positioning during the assembly of the components .

The mechanical part influences the environment and so it can be perceived as a set of effectors. The mechatronic unit gathers data about the environment by using sensors - force and optical, so we assume the unit to be decomposed into three subsystems:
effectors (robot arms,tools and devices) - E ,
sensors - S,
control - Q.

Then the state of the system can be described as:

$$U = \{ \quad E, S, Q \quad \} \qquad (1)$$

where U, E, S, Q are usually expressed as state vectors.

3.3. Modelling of the mechatronic unit

At the modelling the following assumptions are made:
-We considered a robot to be a device, which applies desired contact forces in a controlled way. These forces are specified in three translational and three rotational directions of some cartesian reference frame.

In the design and simulation the main object is to create an interactive package that can be run in tandem with a physical robotic world. The description of the object model is devided into four categories:

A) The dynamic module, related to forces required to cause motion.It includes data such as internal forces, coriolis forces and the kinetic energy matrix.

B) The kinematic module includes data, such as joint angles, velocities, the jacobians and transformation matricies, which can be used to calculate parameters, such as the position and velocity of the robot endeffector

C) The geometric module stores the geometric description of the assembly structures, building the mechatronic unit.

D) The graphical module determines the visual image of the objects in a graphics output.

For modelling and simulation it is necessary to compute joint acceleration from joint positions and velocities from the actuator torques
or forces. The control algoritms also involve the computation of the dynamic model.The common base for this is the Newton - Euler algorithm(formalism), which implies two recursive computations :

a) forward recursion from the base to the robot endeffector (computing the velocity and acceleration of each link),

b) backward recursion from the effector to the robot base (computing the forces and torques to each joint).

For fast and accurate motions of main and assisting robots is necessary to compute in real time the controlled torques as functions of the coordinates.

Taking into account the force and moment , acting on the endeffector of the main robor M_r - F_{as} , n_{as} (assembly force and moment) as initial conditions, we obtain an effective algorithm for the modelling of two robots, as a dynamical subsystem of the mechatronic assembly unit.

3.4. Multiprocessor control and integration

As we notice in our approach to integration of mechatronic systems is suggested to use multiprocessor control system for assembly robots and basic station.In such way is possible to execute several tasks of motion planning and fine positioning simultantny , using so called satelite processors and the interprocessor comunication. In the case of highprecizely assembly operations is necessary to measure on - line a large volume of contact forces, and to perform fine positioning by the basic station, using so called active compliance. (Georgiev,1994)

The direct dynamic model is used to simulate the responce of the robots. Using the equation :

$$T = A.\ddot{q} + B(q, \dot{q})\dot{q} + Q \qquad (2)$$

where : B- represents the centrifugal and coriolis force vector, Q - gravity force vector, A - inertia matrix of the robot, T -torque vector.
We get :

$$\ddot{q} = A^{-1}[T - H(q, \dot{q})] \qquad (3)$$

where : $H(q, \dot{q}) = [B(q, \dot{q}) + Q + J^T.F_t + N]$ (4)

F_f - the friction forces , N - external effort (force and moment)
also can be written :

$$\frac{d}{dt}\begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ -A^{-1}.[H(q,\dot{q})] \end{bmatrix} + \begin{bmatrix} 0 \\ A^{-1} \end{bmatrix}.T \qquad (5)$$

and $y = q$, $\dot{y} = X$.

In this representation the state variables are represented by the vector

$[\, q^T \dot{q}^T \,]$, while $y = q$ gives the output in the joint space, and $y = X$ gives the output vector in the operational space. The calculation of $H(q, \dot{q})$ can be obtained by the Newton - Euler algorithm, noting that $H(q, \dot{q}) = T$, when $\ddot{q} = 0$

4 . CONCLUSION

The multiprocessor control gives a variety of possibilities to determine when the tasks start and when they can be stopped. This was done by integrating classical control structures with reactive ones and synchronisation schemes of the robot tasks.The global control scheme
is based on the knowledge that robot tasks are made of a connected set of elementary tasks (components), for example Jacobian matrix, error vector computation , etc.

REFERENCES

1.Burr J.,(1990) A theoretical approach to mechatronics design, Institute for engineering design, Techn. University of Denmark, Lyngby,.
2.Galabov V., et al., (1992) Mechatronic approach to synthesis of mechanisms in robotics, Proceedings of the 5th Internat. youth summer school "Application of mechanics & biomechanics in mechatronics", Varna
3.Kallenbach E.,(1993), Mechatronic – Systemintegration, Proceedings of the 38 th IWK , Ilmenau, Germany.
4.Georgiev Kr.,(1996), Robotic cell for measurement and adaptive assembly, Proceedings of 6th International symposium "Measurement and control in Robotics ",IMEKO , Brussels,
5.Georgiev Kr.,(1995),Development of a mechatronical adaptive device for assembly robots, Journal of Intelligent Mechatronics, vol 1, No.2, Ankara
6.Georgiev Kr.,(1994), Robotic adaptive unit for precize assembly Proceedings of the seminar of TU-Vienna on "Handling and assembly of microparts", November 1994, Vienna.
7.Featherstone R.,(1987),Robot dynamics algorithms, Kluwer Acad. Publ.
8.Honekamp R.et al.,(1997)., Structuring approach for complex mechatronic systems, Proceedings of ISATA ,Florenz,VI.1997.

Yves Gonthier Yves.Gonthier@space.gc.ca
John McPhee McPhee@real.uwaterloo.ca
Christian Lange Christian.Lange@space.gc.ca
Jean-Claude Piedbœuf Jean-Claude.Piedboeuf@space.gc.ca

A Regularized Contact Model with Asymmetric Damping and Dwell-Time Dependent Friction

University of Waterloo

&

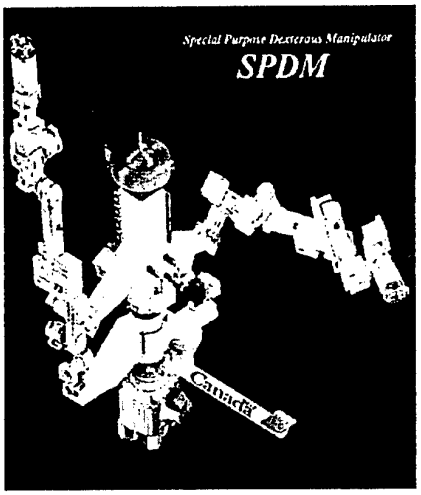
Canadian Space Agency

CSME Forum 2002, May 21-24, 2002 at Queen's University, Kingston, Canada
http://conn.me.queensu.ca/~csme2002

Presentation Outline

- **Motivation**
- **Contact Model: Hertz Theory**
- **New Damping Coefficient Definition**
- **New Friction Model**
- **Simulation Results**
- **Conclusion**

Motivation



- **SPDM Task Verification**
- **High Fidelity Simulation**

Regularized Model: Hertz Theory

- **Solution derived from the theory of elasticity**
- **Stiffness force models for bodies with**
  - **Non-conforming shapes**
  - **Smooth surfaces (i.e. no edges)**
- **For sphere-sphere contact**

$$f_n = k\, x_n^{3/2}$$

Regularized Model: Hunt-Crossley

- **Hertz model + hysteretic damping**

$$f_n = k\,x_n{}^p + \left(\lambda\,x_n{}^p\right)\dot{x}_n$$

- **Can be re-written as**

$$f_n = k\,x_n{}^p\left(1 + a\,\dot{x}_n\right)$$

- **Where**

$$a = \frac{\lambda}{k} = \frac{3}{2}\alpha \qquad e = -\frac{v_o}{v_i} = 1 - \alpha\,v_i \text{ (Goldsmith)}$$

$$\alpha = 0.8 - 0.32 \quad \text{[s/m]}$$

Regularized Model: New Model for a

- **Solve ODE**

$$m\,\ddot{x}_n + k\,x_n{}^p\left(1 + a\,\dot{x}_n\right) = 0$$

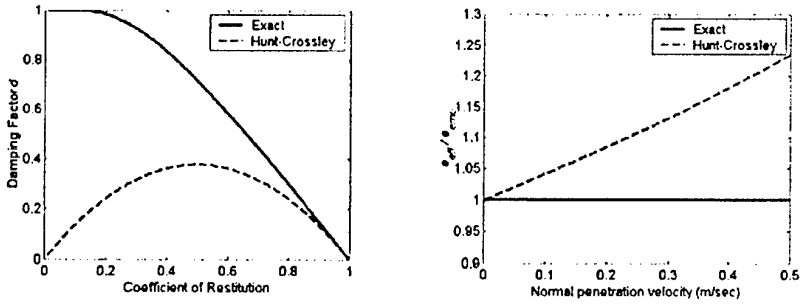
- **Separation of Variables**

$$\int \frac{\dot{x}_n}{1 + a\,\dot{x}_n}\,d\dot{x}_n + \frac{1}{m}\int k\,x_n^p\,dx = 0$$

  - **Solve for a in terms of v_i, v_o**
  - **Independent of stiffness model exponent**
  - **Can also solve for $x_{n,\max}$**

Regularized Model: New Model for a

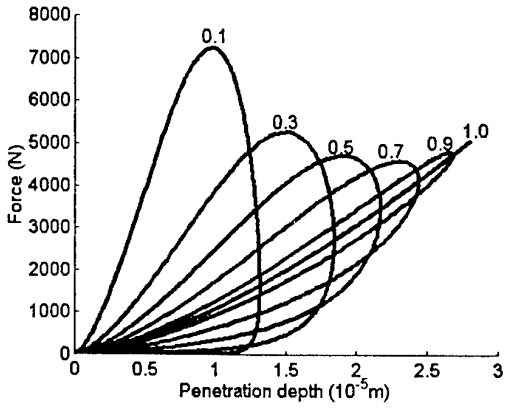
- **Solution:** $a = \dfrac{d}{e v_i}$



– **Parameter d is a dimensionless function of e**

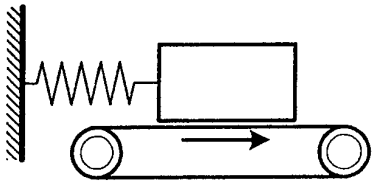
Regularized Model: New Model for a

- **Force response curve**



Regularized Model: Friction Model

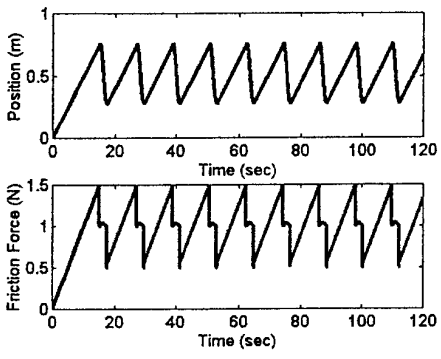
- **Rabinovicz experiment**



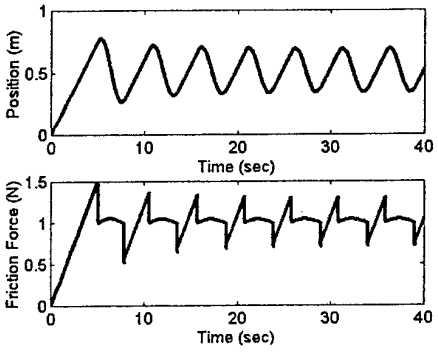
  - **Vectorized bristle model includes:**
    - **Dahl effect**
    - **Stribeck effects**
    - **Stick-Slip Effect**
  - **Dynamic behavior: dwell-time**

Regularized Model: Friction Model

- **Rabinovicz experiment**



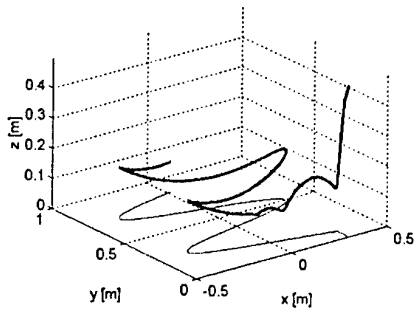
Slow Motion

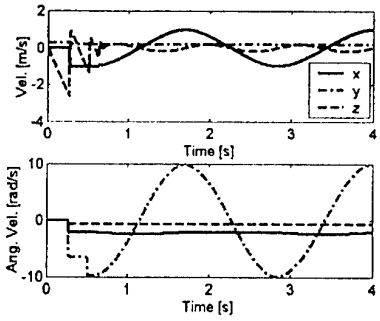


Fast Motion

Hertz Theory: Complete Model

- **Simulation results**





- **Bounce:** $e = 0.5$
- **Sticking and rolling**
- **Friction direction changes**

No rolling or spinning friction!

Conclusions

- **3D Contact model includes**
  - **Normal force and damping**
  - **Tangential friction**
- **New definition for damping coefficient**
  - **Works for high & low coefficient of restitution**
  - **Exact solution of the ODE**
- **New friction model**
  - **Stick-Slip transition**
  - **Dhal and Stribeck effects**
  - **Frictional lag (dynamic friction)**
  - **Resulting ODE is not stiff (can use explicit solvers)**

MULTI-BODY HUMAN MODEL

L. HYNČÍK
New Technologies Research Centre
in the West Bohemian Region
University of West Bohemia
Univerzitní 22
306 14 Plzeň
Czech Republic

1 Introduction

In last decades car industry is quickly developing. That is why a strong emphasis is taken into account for the safety of the traffic. Cars are equiped by safety belts and recently frontal and even side airbags begin to belong to standard equipment of most cars. Those belts and airbags must be validated by experimental tests which are expensive. To exchange those tests by simple calculation was just couple years ago impossible because of considerable extensive task. By intense development of the computers is this task possible today. From the point of mechanics an available method is chosen and a numerical model is implemented by computer. The work shows development of the numerical human model which is necessary to improve the car safety in comparison to the dummy numerical models which exist since more time but they are not enough "biofidelic" to represent the human behavior.

2 Objectives

The aim of the human articulated multi-body model was to divide the human body model given by meshed geometry into the rigid bodies corresponding to the real human body parts including inner major organs and tissues optionally, to give them right physical properties, i.e. masses and inertias, and connect them with well calibrated joints and contacts. Furthermore a mobile shoulder joint was added and improved including passive muscle bars. The human articulated rigid body model is structured like existing articulated rigid body dummy model, *HYBRID III 50%* [4]. It uses anthropometric data found in the literature [3]. The model is based on the finite elements under the *PAM-SAFE* ^{TM} system.

3 Basic Human Model

The meshed geometry of skin, skeleton and the surfaces of inner organs and tissues (optionally) are taken into account based on [5] kindly provided by the ESI company. The skin has the importance as the contact surface when running the sled test. The skeleton is used rather more for the visual effect, however it helps to deduce the approximate joins locations. The organs can predict acceleration inside during any human activity and serve as a background for more elaborated deformable models. The anthropometric data other than skeletal and outer skin geometries, such as segmentation into 15 segments with 9 different segmentation planes, 10 different segment origins and coordinate systems, segment centers of gravity and principal inertia axes, segment masses and inertias and joint locations and axes, ranges of motion and resistance of motion for a mid-sized male human were taken from [3]. The skeletal and muscular anatomy of the added shoulder joint, including the clavicula, scapula, thorax and humerus bones geometry, joint locations, excursions and orientations, as well as the muscles that connect the mobile shoulder parts, the scapulae, to the trunk and to the upper arm, are based on data found in Kapandji [2] and in Gray's anatomy [6].

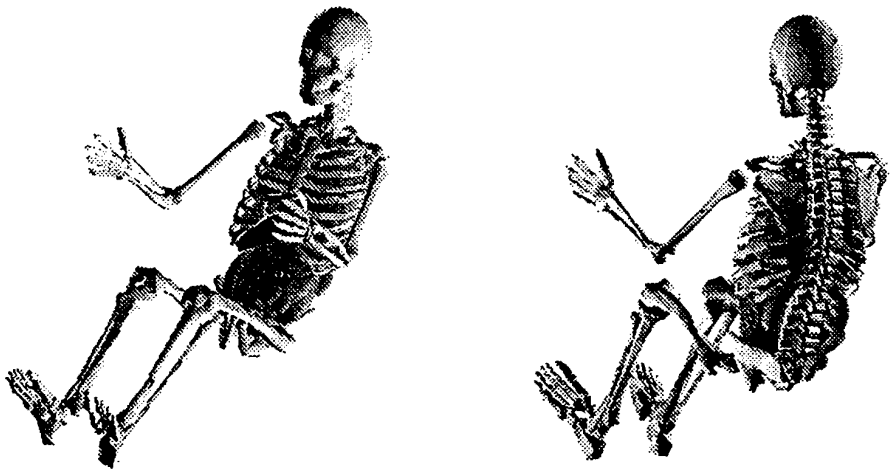


Figure 1: Car sitting human model

To detect dynamical properties of inner organs and tissues, a simple experiment on a mid-sized cadaver was done. After weighting and volume measuring of particular organ or tissue, uniform density was computed. The density was a input to the Gauss-Ostrogradsky's theorem for dynamical properties computation. The theorem was used because of the surface mesh of inner organs and tissues.

All materials except muscles are simplest materials without any resistance since the existence of rigid bodies. In order to provide a continuous outer skin contact surface under articulated motion, certain sets of facets were introduced.

Those facets do not belong to one single rigid body but they connect corresponding nodes of skin portions between different articulated rigid body parts. That permits the nodes of these facets to move freely with the respective rigid body to which they are attached while the connected facets bridge the gaps between the skin of adjacent articulated members and can serve as continuous freely deformable contact interfaces.

4 Multi-Body System

The human geometry has been divided into 9 parts, namely head, neck, upper part of body, lumbar spine, lower part of body, left arm, right arm, left leg and right leg. The arms and legs contain subparts, namely lower and upper arms, hands and lower and upper legs, feet, respectively. The basic model contains 21 rigid bodies corresponding to parts or subparts mentioned above, namely head, neck, upper part of body, lumbar spine/abdomen, lower part of body, left and right clavicula, left and right scapula, left and right upper arms, left and right lower arms, left and right hands, left and right upper legs, left and right lower legs and left and right feet. The centers of gravity, inertia axes, masses and principal inertia axis and principal moments of inertia were obtained from [3]. There is a contact defined between bodies which are supposed to contact during motion.

Concerning inner organs and tissues, there are brain, larynx, trachea, lungs, heart, diaphragm, liver, spleen, gall bladder, stomach, intestines, kidneys, adrenal glands, urinary bladder and prostate modelled as 18 particular rigid bodies. They are connected by soft and tied conntacts. In order to have proper dynamical characteristics of the outer body parts, they were corrected based on dynamical characteristics of inner rigid bodies.

5 Modelling of Joints and Ligaments

For the location of joints the dummy described in [3] was used. From the *HYBRID III 50%* data set, the method defining the local frames was used. The kinematic joints provided by the *PAM-SAFE* ™ solver were used for added CPU efficiency. The shoulder complex combines two kinds of joints, namely

- anatomical joints, which are defined as joints between two bones, included in shoulder, which are controlled mainly by ligaments (the gleno-humeral, sterno-clavicular and acromio-clavicular joint),
- physiological joints, which are defined as joints between two bones or connected parts without capsula and ligaments and which are controlled by contact surfaces, sliding interfaces, ligaments, tendons and muscles (the scapulo-thorasic joint and the secondary subdeltoidal, costoclavicular joints).

The anatomical joints can be modelled with a numerical joint element group up to 6 degrees of freedom of which are brought in correlation with its anatomical capacity of motion. We used 3 rotational degrees of freedom. The model contains 4 flexion-torsion joints situated in the spinal column and 16 spherical joints in shoulders and extremities. It is assumed that their centers of gravity are fixed. The different properties of the anatomical joint such as the elasticity of the ligaments and capsulae, values for friction, cartilage, damping etc. are taken into account in the material properties of the numerical joint elements.

The physiological joints are controlled by sliding interface surfaces and by passive and/or active muscle action. For that joint type muscle modeling becomes important and necessary to simulate their kinematics, resistance and excursion. The subdeltoidal joint being just a subsidiary joint in relation to the main gleno-humeral joint is not represented in the model. On the other hand, the scapulo-thorasic joint is modelled in detail with sliding interfaces situated between the anterior surface of the scapula and the posterior surface of the rib cage and between the posterior surface of the scapula and the dorsal skin inner surface and with muscle bars for all muscles which link the shoulder complex to the trunk.

The ligaments are modelled using tied contacts between particular organs and tissues. Soft contacts are added to avoid mutual penetration between particular organs and tissues.

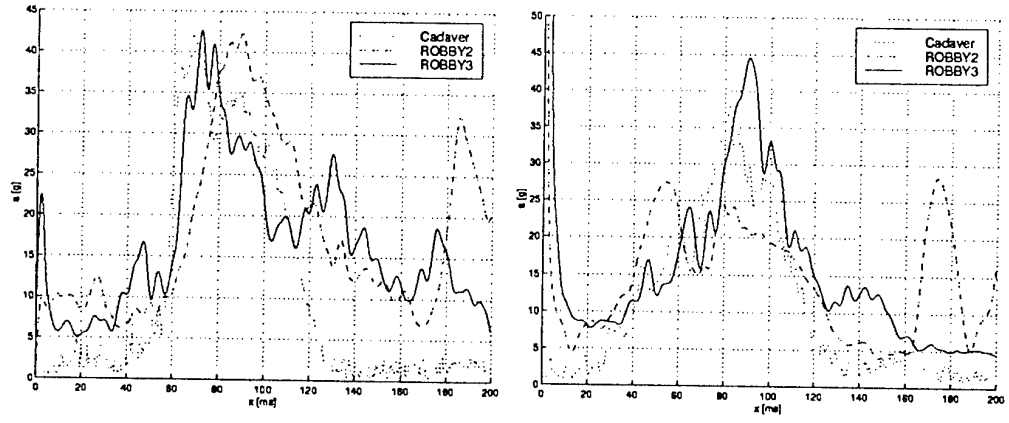


Figure 2: Head and thorax accelerations compared to experiment

6 Validation

The validation is based on a test conducted by Kallieris [1]. The model is seated on a rigid seat and restrained by a three point belt system and an airbag. The car interior parts (cushion seat, back seat, floor panel, foot rest, instrumental panel, steering and wind screen) are modelled as flat fixed surfaces contacted with body parts. The car interior matches the BMW series 3 car. The belt

system is modelled as assembly of bars. Airbag is modelled as a standard model in the *PAM-SAFE* ™ system. Acceleration measured during experiment was applied to all centers of gravity.

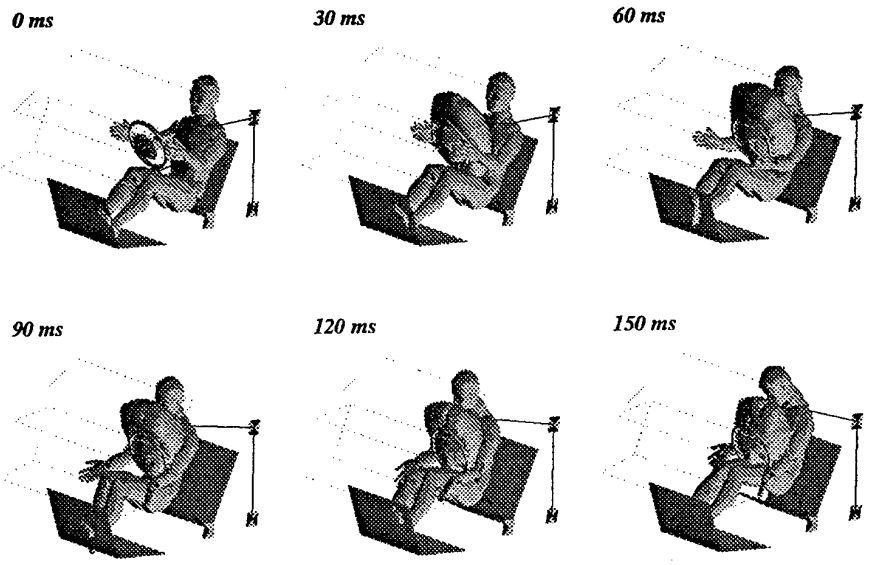


Figure 3: Sled test computational results

7 Conclusion

The model can serve as a basic model for more elaborated human articulated rigid body models. The human articulated rigid body model can also serve as the locally detailed finite element model of various body parts. If grafted onto the human articulated rigid body models, the refined parts can be studied under realistic kinematic boundary conditions as they result from the overall kinematic motion response of the body.

8 References

[1] Kallieris, D.: *Three Cadaver Tests by using 3-pt-belt - Force Limiter - Airbag Combination*, A Report for National Highway Traffic Safety Administration, Contract No. DTNH 22-89-D-05012, 1993

[2] Kapandji, I. A.: *Physiologie Articulaire*, vol. 1 - 3, 1994

[3] Robbins, D. H.: *Anthropometry of Motor Vehicle Occupants*, vol. 2 - 3, The University of Michigan, Transportation Research Institute, UMTRI-83-53-2, 1983

[4] Users' Manual of the 50% HYBRID III Test Dummy, *SAE Engineering*, 1985

[5] ViewPoint Catalog, Summer '96 Edition, *External and Skeletal Anatomy*, 1996

[6] Williams, P. L.; Warwick, R.; Dyson, M.; Bannister, L.: *Gray's Anatomy*, 1989

[7] Winters, J.; Stark, L.: Estimated Mechanical Properties of Synergistic muscles Involved in Movements of a Variety Human Joints, *Journal of Biomechanics*, **vol. 21, N° 12, pp. 1027-1041**, 1988

9 Acknowledgments

This work is supported by contribution of MŠMT under research and development project LN00B084. Special thanks belong to Biomechanics team of ESI Group for cooperation and to Heidelberg University and BMW for experimental data.

10 Information

LUDĚK HYNČÍK
New Technologies Research Centre in the West-Bohemian Region
University of West Bohemia
Univerzitní 8
306 14 Plzeň
Czech Republic
Telephone: +420–19–7491586; Fax: +420–19–7421420
E-mail: hyncik3@ntc.zcu.cz

DYNAMICS OF VIBRATING SYSTEM WITH ACTIVE CONTROL

S. F. JATSUN, *A.S. ZAISEV, * S. M. JATSUN **

*Kursk State Technical university
305040 50 Let Oktyabrya, 94
**Kursk State Pedagogical University
305000 Radisheva st. 33

1. Introduction

The analysis of dynamics of an electromagnetic vibrating drive with a control system is represented. The behavior of a system is shown at a modification of magnitude of an exterior force. Vibrating technologies is widely used in different industry applications [1,2,3].
The research of dynamics is based on the diagrams of bifurcation and Lissajous curves. With introduction of a loop of a feed-back the amplitude of oscillations of the executive link of a vibrating drive is supported at a constant level at a modification of a technological force. The same behavior of a system of an electromagnetic vibrating drive is characteristic at magnification of an exterior force up to an installed maximum value obtained in an outcome of numerical simulation. At as much as possible maximum safe loads, are observed vibration shock modes of operations.

2. Mathematical model

In this paper we investigate electromagnetic vibrating drive with nonlinear elastic suspension and system of active control of vibrating motion. As usually if we use passive drive the amplitude depended on forces that act on the executive element of drive When the loading increases the amplitude go down. Sometimes such process is not acceptable. For decision of this problem we use active feed back control system. The scheme of drive is shown on fig.1. on the scheme we can see vibrating mass m, nonlinear suspension with corresponding reological parameters. General coordinate X describes motion of drive. Electromagnetic force F acts on the mass from side of electromagnetic coil. Sensor D measure acceleration and velocity of mass and gives information to the control system for modification of electrical feeding.
For description of dynamic of motion of this electromechanical system we use Lagrange-Maqswell equations.

$$\begin{cases} m\ddot{x} + \beta\dot{x} + cx = F_{эл-магн.} \\ \dot{\Phi} + R\dfrac{\Phi(\lambda \pm x)(2 + K_S)}{\mu_0 Z^2 S} = U \end{cases} \qquad (1)$$

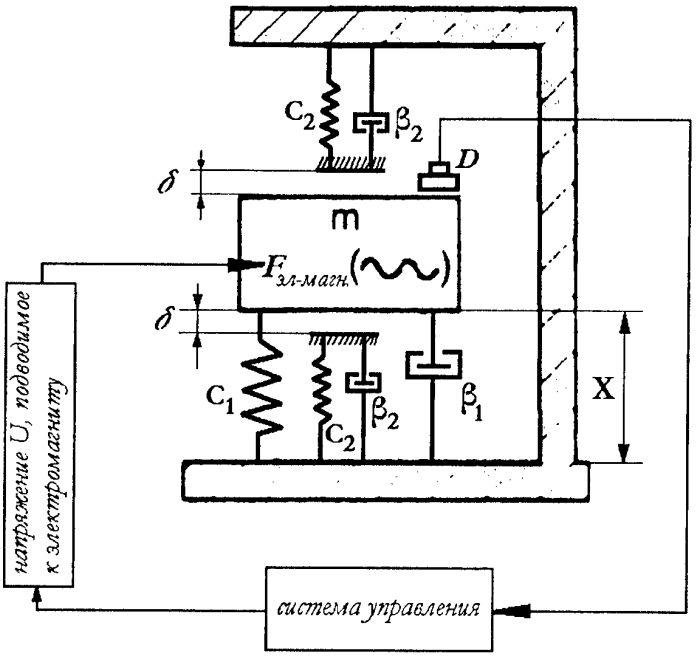


Figure1. The scheme of the one mass electromagnetic

Were $F_{эл-магн.} = \frac{-\Phi^2(2+K_S)}{\mu_0 Z^2 S}$ –acting force of electromagnet; β- Viscosity of coefficient. c – stiffness coefficient; Φ –magnetic flax; K_S –koefficient which take into consideration geometrical parameters of coil; λ –air gape between anchor and stator; R –active resistance; μ_0 –magnetic constant; S –cross section square; Z –number of circles of coil; $U = f(t, \ddot{x})$ – electrical feeding depends on time and acceleration of executive element. Fig.2 are shown characteristic of elastic force in dependence on displacement.

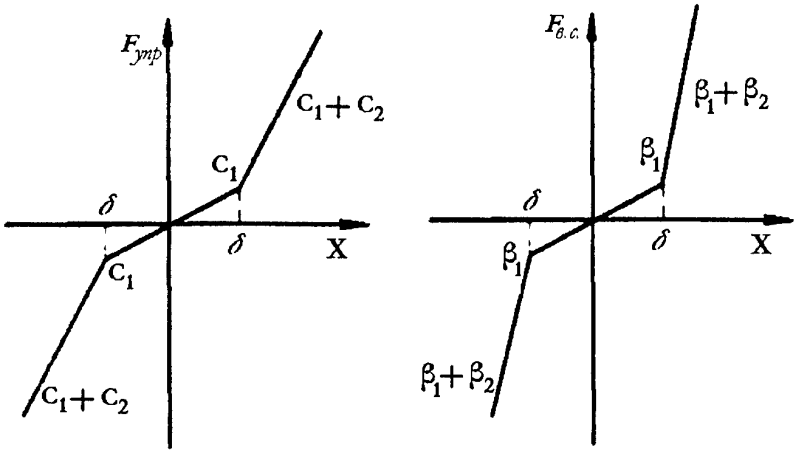


Figure 2. Characteristic of elastic force in dependence on displacement

The system moves with accordance of next procedure: mass m vibrate under action of the electromagnetic force. Sensor D is placed on the executive element. Signals from sensor came to the control system. We consider very important for practis case when amplitude of executive element should be constant and independent on external forces.
Let electromagnetic force depends on time with accordance of a formula:

$$U = U0+U1\sin(kt) \tag{2}$$

Where, U0-constant strain of feeding, U1-amplitude of changeable feeding.
In this case, we have two parameters of regulation U1 and U2.
During increasing of forces on executive element , where sensor is placed, amplitude of executive element decreases. The control system measures level of middle acceleration and changes the parameters of electrical feeding of electromagnet. We investigated different control strategies. One of them is shown below. Criterion of quality of control system in this case is the function F-minimum of differences between curves on fazes square which we got for various loadings.
Actually we should determine of the lows of changes of regulation parameters U0,U1 in dependence on middle acceleration a=1/T.

$$U0=U1(a) \tag{3}$$
$$U1=U2(a) \tag{4}$$

For approximation of formulas (3),(4) are used linear:

$$U=ba+c; \tag{5}$$

where U={U1,U2}, b={B1,B2},c={c1,c2}
and non linear strategy:

$$U=c+ba+Da^2 \tag{6}$$

where D={D1, D2}
We should to determine parameters of vectors b, c, and D, which minimize the function of mistakes F. For decision we used the technology of multi dimensional sound probes in the space of changeable parameters b, c, D. On this stage we got analytical dependence of function:

$$F=F(b,c,d), \tag{7}$$

and after getting of a response surface F(b,c,D) we solved a problem of nonlinear programming, when we determined parameters b,c,D, which provide minimum of F- function.

3. Results of calculation

On the fig.3 is shown results of investigation and dynamical response of system in a case, when loading is absent. The control system if switched off U1=0,U2=0.In this case we have harmonic motion of drive with constant amplitude without of impacts.
On the fig.4 is shown the motion of drive, when control system switching on but U1=0, only one part of regulation strain can change. External force changes in interval from zero to 30N. Analysis of fig.4 shows that during change of loading in interval from P=0 to P=30N

amplitude of displacement of executive element practically is constant. At these time increasing of electrical feeding U2 brings constant component to electrical strain and as consequence lowering of air gape in electromagnetic part of drive, that provide appearance of impact regimes, as it is shown on the fig.4 and fig.5.

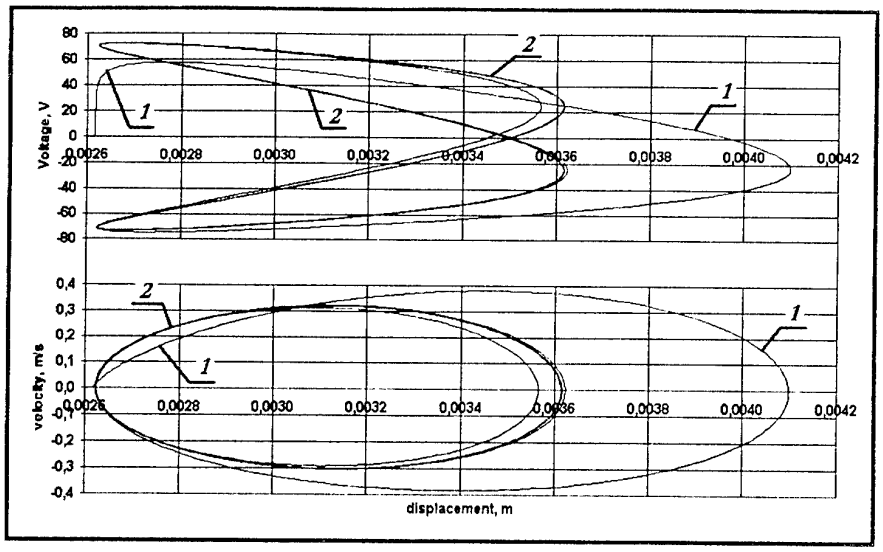


Figure 3. Faze characteristic of electromagnetic drive without loading
1 –the moment of time when electromagnetic drive turn on;
2 – stabile regime of motion.

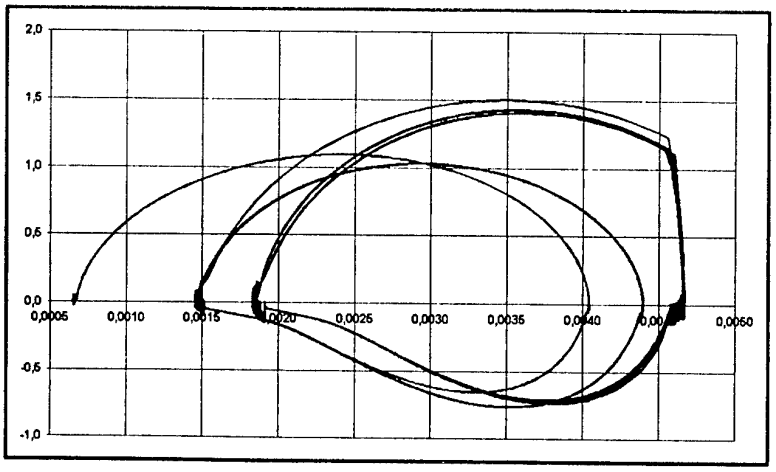


Figure 4. Fazes characteristic of electromagnetic drive,
when control system is switching on
regulation U1=0,U2=U2(a)

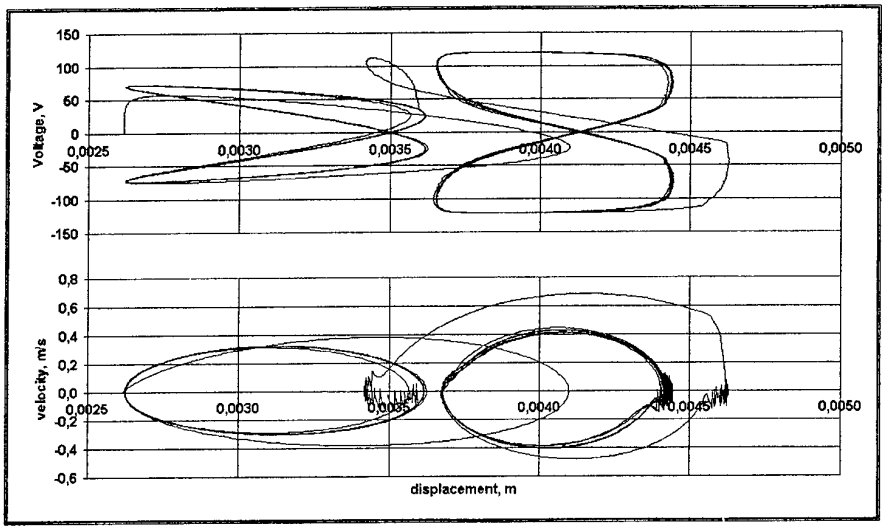


Figure 5. Fazes characteristic of electromagnetic drive with control system

For exception of displacement of zero (statically position of executive element) we used full regulation, when U1=U1(a) and U2=U2(a). On fig.6,7 is shown results of calculation of dynamical motion of system while full control system switching on. We can see that this regulation provides practically full independence of parameters of vibration on external forces.

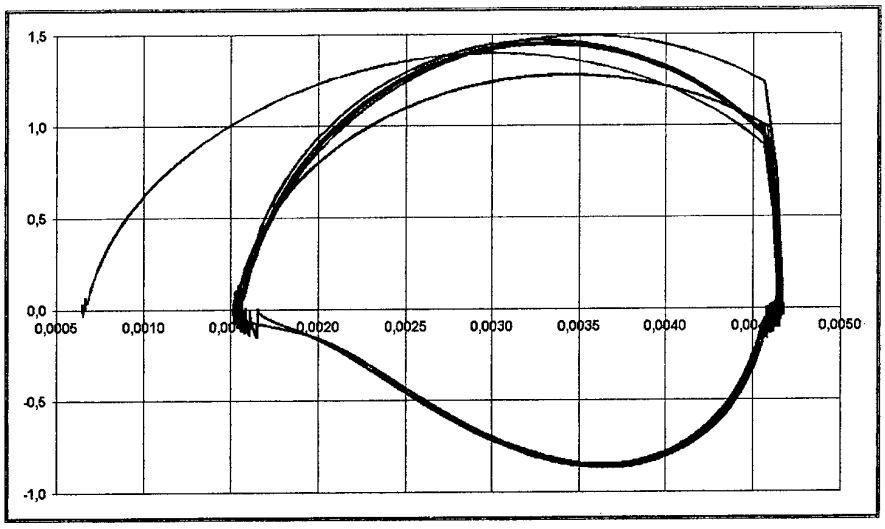


Figure 6. Fazes characteristic of motion while full regulation U1=U1(a),U2=U2(a).

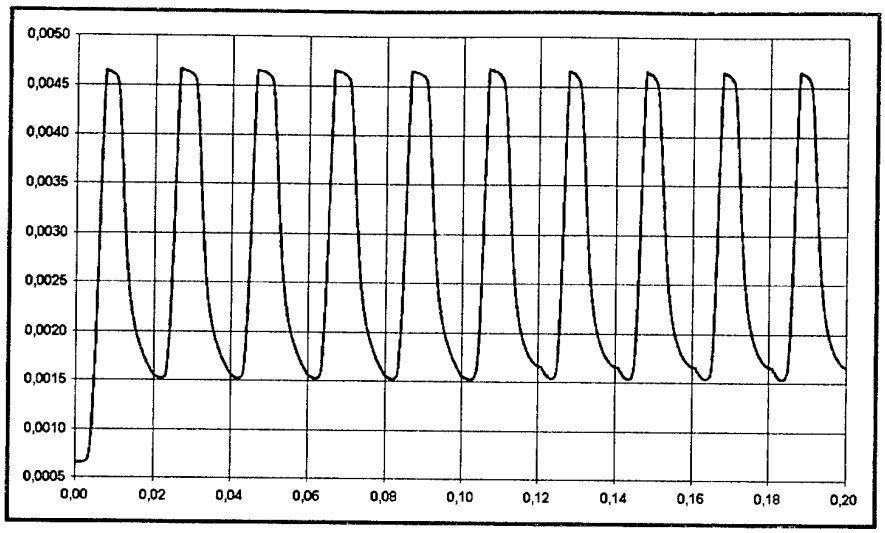


Figure7. Motion of drive in a case when full control system is switched on U1=U1(a),U2=U2(a).

4. Conclusions

We create the mathematical model, which describe dynamical behavior of electro mechanical system with feed back control system. Different control strategies are investigated. The results of calculation show that only two dimensional regulation, when U1=U1(a) and U2=U2(a) provide practically independence of vibrating parameters (amplitude of displacement, acceleration, velocity) on external forces.

5. References

1. Jatsun S., Jatsun S., Shepoluhin V. (2001) Vibrating method for measurement of the biomechanical properties of human skin, Proceedings V International congress of sound and vibration, 1216-1227.
2. Yatsun S., Safarov I.(2000) Dynamics of vibrating engine for walking spain robots. CLAWAR. Madrid, 529-534.
3. Yatsun S., Safarov I., Vorontsov R. (2001) Dynamics of robot with vibrating engine. CLAWAR, Karlsruhe German, 1150-1156

DEVELOPMENT OF IMPEDANCE CONTROL METHOD FOR MECHATRONIC SYSTEMS

K.Gr. KOSTADINOV, G.V. BOIADJIEV
Mechatronic Systems Dept., Institute of Mechanics
Acad. G. Bonchev St., Block 4, Sofia 1113, Bulgaria

1. Introduction

Various force control methods have been developed to meet the requirement of regulating the contact force within a specific range. The force control is based on two distinct methodologies: pure force control and impedance control (Fig.1.). Pure force control can be applied only when the end-effector is in contact with the environment. While in impedance control the force is regulated by controlling the position and its relationship with the force, i.e. robot mechanical impedance. The impedance control can be realised by three approaches [1]. A lot of works developed the first approach for impedance control by impedance controller using one of the seventh approaches [2]: constant PD control, model based computed torque control, adaptive control, robust saturation-based control, sliding mode-based impedance control [3], learning impedance control [15] or quaternion-based impedance controller [16]. Generally the developed impedance controllers are characterised with the different shortcomings (Fig.1.) which determine the specific practical applications of the first approach for impedance control.
The second approach for impedance control is realised by redundancy of joints. There is limited information for development of this approach, so this paper will be an attempt to induce collaboration in this field.
The second approach for impedance control, i.e. by redundancy of joints, is reduced for one joint into the third approach for impedance control realised by redundancy of actuators for each robot joint. Many researchers developed the third approach for impedance control used antagonistically driven robot joints by two actuators via tendons [4,5]. Antagonistic stiffness, for which the modelling procedure for a completely general kinematic system along with a stiffness formulation technique developed [6], seems to be very unique and promising to design and control the robots and mechatronic peripheral devices with high precision requirements under various operational impacts and disturbances.
This paper considers the development of the impedance control method, especially the second and third approaches as a way to adapt dynamic behaviour of the mechatronic system during its interaction with technological environment in order to improve the process quality and to achieve more system functionality.
The paper is structured into 5 parts. The first part considers the preliminary investigations of the second impedance control approach. The impedance-controlled actuators with drive redundancy are developed in the second part. Experimental results and discussions is subject of the third part.

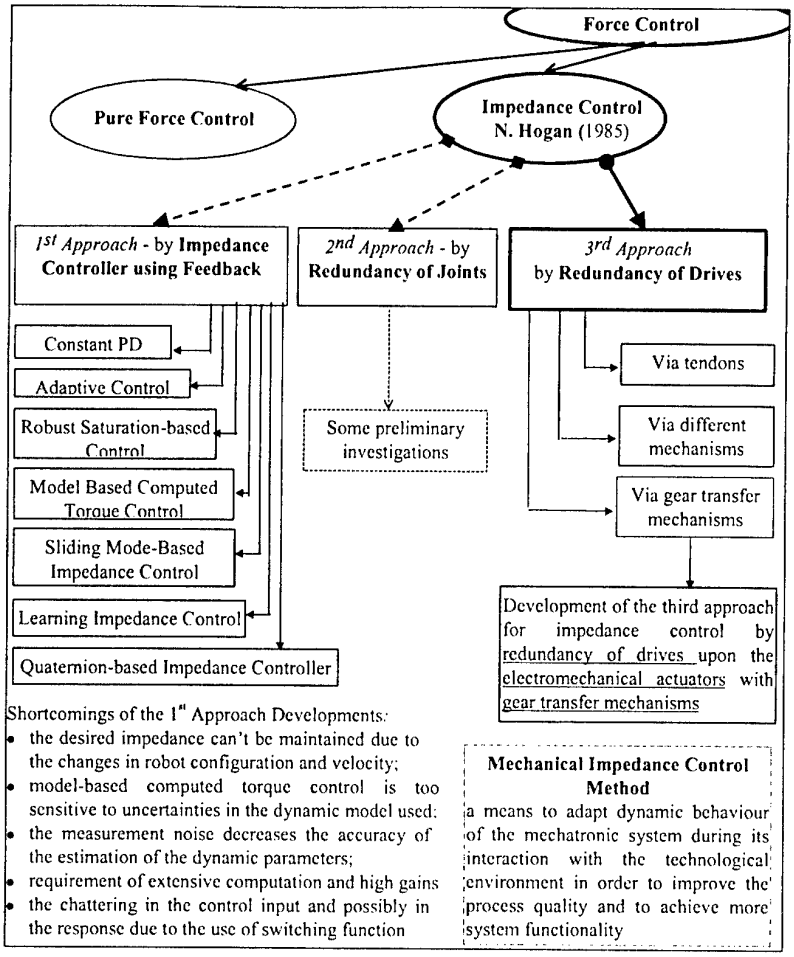


Figure 1: Force Control Methods - Developments of the Impedance Control Method

2. Second Approach for Development of Mechanical Impedance Control.

The second approach for impedance control is realised by redundancy of joints. To apply this approach first is necessary to determine the number of redundant joints for the set of reference tasks. The second one is how to distribute the control of joints in regarding of the kinematic [13] and dynamic sensibility [14] and the task to be performed.

Let's consider the local structure of robot system that performs technological motions in a plane. In tangent direction of the plane trajectory a robot is necessary to control its dynamic properties, i.e. to realise impedance control on that direction. In general case 2 DOF local structure is necessary for this plane task function. But for the case considered we need 2 independent robot joints to set up two control sets [1]: the flow source $\{S_f\}=V_0(t)$ and the actuator mechanical impedance Z - $\{S_Z\}: Z_0(F_{int}, Q)$, where V_0 is end-effector velocity, F_{int} - force interaction, Q - process quality requirements, to realise the mechanical impedance control in that direction. So, it is necessary to have a robot structure with

minimum one DOF more than the task function. Kinematic chain with three revolute joints whose axis are placed perpendicular to each other, i.e. R⊥R⊥R is taken here as an example. Let's consider how can be realise reference trajectory by the second approach for impedance control in a simple case of a plane trajectory motion with targeted impedance in one direction only. The determining of the vector of orientation error $\dot{\theta}(q)$ for this structure where q_i -general coordinates, ψ - angle around the unit vector $\dot{u}$ [14] is by:

$$\dot{\theta} \approx \dot{u}\psi = \begin{vmatrix} \frac{1}{4}[\cos(q_1 - q_2) - \cos(q_1 + q_2) - \cos(q_2 - q_3) + \cos(q_2 + q_3)] \approx 0 \\ \frac{1}{4}[-\sin(q_2 - q_1) - \sin(q_1 + q_2) + \sin(q_2 - q_3) + \sin(q_2 + q_3)] \approx q_2 \\ \frac{1}{2}(1 + \cos q_2)\sin(q_1 - q_3)] \approx q_1 - q_3 \end{vmatrix} . \quad (1)$$

But from the other side: $\delta\theta = L(q)\delta q = \left(\frac{\partial(\psi u)_i}{\partial q_j} \right)_{i,j=1}^{3}$ (2)

So, the matrix L(q) is obtained as simple matrix in first approximation:

$$L(q) = \begin{bmatrix} \frac{q_2}{2} & \frac{q_1 - q_3}{2} & -\frac{q_2}{2} \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix} \quad (3)$$

Here it has to be pointed out two specific features of the matrix L(q): two rows where are placed numbers only; one row with zeroes, i.e. its rank always is smaller then three. So, the rank of L(q) is as follows: rank L(q)=2 at $q_1 \neq q_3$, q_2 - any angle; 1 - at $q_1=q_3$, q_2- any angle. The last means there exist not just isolated points or configurations but whole trajectory we are looking for where distribution of orientation vectors of errors is one dimensional field [13], which coincides with the direction of eigenvector corresponding to this positive eigenvalue. It is just one in this case. The coefficients of dynamic sensibility depend only of values of q_2 when the system follows the way $q_1=q_3$ during orientation like that. In the case of just one non zero eigenvalue it is not difficult to find the corresponding eigenvector. It has the form $\dot{\theta} = [1,0,-1]^T q_1$ (4)

Here $[1,0,-1]^T$ is a fundamental solution of the system for eigenvectors with q_1 as a parameter. And the last result is the same for whole trajectory $q_1 = q_3$.

On the fig.2. is shown a geometric interpretation of a field (plane a.) where the only one eigenvector is contained and q_i(1=1,2,3) are passed through a unit cube. That field is a diagonal intersection of the cube containing the diagonal $q_1 = -q_3$. Therefore, for every force action which is placed in this field (Fig.2.), there exists and can be found such a configuration, where the vector $\dot{\theta}$ will be collinear with it. That means if we have a force then there is no additional compensation of energy has to be given by drives of the system. But if we have a moment then we need additional energy compensation, which is distributed with equal value for q_1 and q_3 and the concrete q_2 which assure colinearity between the outer moment and the vector $\dot{\theta}$. For all vectors perpendicular to the field or the part of them whose are projections on this directions the orientation error does not exist and corresponding coefficient of dynamic sensibility is zero. So, we can do it for arbitrary force or moment action by projection of the vector describing themselves on two perpendicular direction and the conclusions just mention above are still hold.

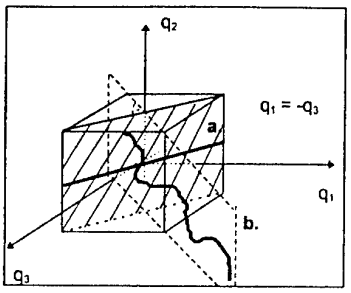


Figure 2. Geometrical interpretation (a.) of the distribution of eigenvector corresponding to one eigenvalue and task plane b.

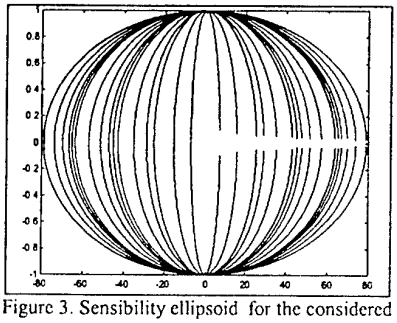


Figure 3. Sensibility ellipsoid for the considered structure R⊥T⊥T

As another example let's consider the planar structure R⊥T⊥T. Each degree of freedom can be taken as redundant one in arbitrary case. The sensibility analysis shows the following results for corresponding sensibility coefficients λ_i,(i = 1,..,3) and directions $X^{(i)}$, (i = 1,..,3), where h_1, p_3 are geometrical parameters.

$$\lambda_1 = 1, \qquad \lambda_2 = 0, \qquad \lambda_3 = 1 + q_2^2 + (h_1 + p_3 + q_3)^2$$

$$X^{(1)} = \begin{bmatrix} 0 & \dfrac{q_2}{h_1 + p_3 + q_3} & 1 \end{bmatrix}^T, \quad X^{(2)} = \begin{bmatrix} \dfrac{1}{q_2} & -\dfrac{h_1 + p_3 + q_3}{q_2} & 1 \end{bmatrix}^T, \quad (5)$$

$$X^{(3)} = \begin{bmatrix} -\dfrac{(h_1 + p_3 + q_3)^2 + q_2^2}{q_2} & -\dfrac{h_1 + p_3 + q_3}{q_2} & 1 \end{bmatrix}^T$$

The second coefficient that is zero assures non sensibility on the direction collinear to $X^{(2)}$ during the end-effector motion. On the remaining two orthogonal directions the sensibility is fixed or variable respectively and following the third one the sensibility depends on the robot configuration. It can be realized separately by one generalised coordinate or by both of them simultaneously. For example the robot deburring task is considered. The end-effector moves on the surface to be deburred. Hence, the non-sensibility direction coincides with the surface normal vector. For the remaining directions the robot has some sensibility according to the executing trajectory. In our example the reference deburring trajectories having these characteristics are arbitrary. If that additional requirement appears, the problem of obtaining the desired sensibility arises which could be solved using structures with redundancy DOF. The sensibility ellipsoid is visualised (Fig.3.) for the considered structure in a limited region of variation of the generalised coordinates.

3. Impedance Controlled Mechatronic Actuators with Drive Redundancy.

The actuators with drive redundancy are a simplified model of the antagonistic drive in the living nature. Actuation and kinematic redundancy are the means by which living species control the dynamic response and modulate their end-effector stiffness.

The synthesis of actuators with drive redundancy is based on the motion transition method [8]. It consists of two-zone dynamic controlled gearing on the actuator output link through the introduction of an antagonistic drive unit, that is identical to the existing one. This method also allows eliminating or reducing the influence of uncertainties on the kinematic chain of the mechatronic drive due to closed loop structures with the additional drive torque. Redundancy actuation causes internal force/torque in the transfer mechanism. This

torque does not perform any effective work to the external world. But the actuator joint stiffness and damping depend on this internal torque [9-10]. In such way the knot mechanical impedance Z_0(K,B) [1] can be controlled. Torque of interaction T_i between the mechatronic system output link and technological equipment can be expressed as:

$$T_i = T(\varphi,\dot{\varphi}) - J\frac{d\dot{\varphi}}{dt} = K(\varphi_r - \varphi_a) + B(\dot{\varphi}_r - \dot{\varphi}_a) - J\frac{d\dot{\varphi}}{dt} \quad (6)$$

where $T(\varphi,\dot{\varphi})$ is knot component of the mechatronic system impedance. The dynamics of the end-effector of the technological equipment, when it is on ideal rigid body, is:

$$J_\Sigma \frac{d\dot{\varphi}}{dt} = T_e + T_i \quad (7)$$

were J_Σ is inertia tensor of the technological equipment end-effector; T_e - unknown torques and impacts. The motion equation of the system "mechatronic system actuator joint shaft - technological equipment" is:

$$T_i = T(\varphi,\dot{\varphi}) - J\frac{d\dot{\varphi}}{dt} = K(\varphi_r - \varphi_a) + B(\dot{\varphi}_r - \dot{\varphi}_a) - J\frac{d\dot{\varphi}}{dt} \quad (8)$$

For the feeding operation is most important to assure smooth motion, i.e. $d\dot{\varphi}/dt \approx 0$. The equation (6) will be:

$$(J_\Sigma + J)\frac{d\dot{\varphi}}{dt} = K(\varphi_r - \varphi_a) + B(\dot{\varphi}_r - \dot{\varphi}_a) + T_e - J\frac{d\dot{\varphi}}{dt} \quad (9)$$

The equation (9) means that by variation of the actuator knot mechanical impedance Z_0 to reject the disturbances at some impacts. The desired actuator shaft response $\varphi_a(t)$ and $\dot{\varphi}_r$ to the reference motion φ_r and $\dot{\varphi}_r$ and the external torques T_i is defined by (6).

Accommodation control of the dynamic accuracy [7] is used. The control scheme consists into 2 part. First one is a feed forward controller, constructed off-line using the obtained dynamic model [11]. The open-loop impedance control involves off-line planning of the targeted actuator mechanical impedance for the desired output link velocity $\dot{\varphi}_r(t)$, smoothness of motion $\Delta\dot{\varphi}(t)$ and expected impacts and disturbances T_i, determining the actuator control sets $\{S_f\}$ and $\{S_z\}$. This allows open loop disturbances and impacts rejection. The second part is a feedback controller used to compensate on-line small perturbation of the expected impacts and non modelled dynamics. Hence, the proposed control strategy for control of actuators with drive redundancy does not need large computational resources and time in on-line control, while adjusting the knot mechanical impedance of the actuator allows open-loop impacts and disturbances rejection. It also does not have a time delay, which often deteriorates performance of rapidly changing processes.

4. Experimental results and discussion

Three experimental test beds with drive redundancy have been designed to investigate the redundancy actuators with impedance control:

- rotary table for feeding operations in robot-assisted material removal [10];
- barrier actuator for a Langmuir - Blodgett monomolecular film deposition system [10];
-harmonic actuator for robot manipulators and peripheral mechatronic devices [12].

The main features of impedance controlled actuators with drive redundancy consist into:
- controlled modification of the actuator mechanical impedance[10];
- micro motions of the barrier - particularly 1μm, at a range of motion (350 mm);
- smooth motion in a wide technological max-to-min velocity ratio (1:10000) regardless of the force interaction at adjusting the impedances of actuator and technological equipment.
- significant linearization of the actuator transfer mechanism [10].

5. Conclusions and future work.

Three impedance controlled actuators with redundancy have been built in the Mechatronic Systems Department as experimental test-beds for various ongoing research activities including mechanical and control design, motion control and for calibration procedure synthesis purposes. Based on the third approach for impedance control positioning robots and peripheral mechatronic devices with such redundant actuators can accomplish very fine motion regardless of their dynamic interaction with remaining robotized equipment.
The second approach for impedance control is under development. The study has to be taken into account as the phase of conceptual design of such mechatronic systems. The reference task is considered in the terms of kinematic and dynamic sensibility. The number of joint redundancy has to be determined on that base. As an aplication the redundancy influence on the mechatronic device for driling operations is under investigation.

Acknowledgements: The authors gratefully acknowledge partial support from NSF under the Project **TH-708/97**.

References:

1. Hogan, N. (1985) Impedance control: An approach to manipulation, Trans. ASME Jour. *DSMS* vol.1, pp.1-23.
2. Lu, Z. and Goldenberg, A. (1995) Robust Impedance Control and Force Regulation: Theory and Experiments, The Int. J. Of *Robotics Research*, Vol.14, No.3, pp.225-254
3. Lu, Z., Kawamura, S. and Goldenberg, A. (1995) An Approach to Sliding Mode-Based Impedance Control, IEEE Transactions on *Robotics &Automation*, vol.11, No.5, pp.754-759.
4. Jacobsen, S.C., Ko, H., Iversen, E.K. and Davis, C.C. (1989) Antagonistic Control of a Tendon Driven Manipulator, in Proc. of the IEEE Int. Conf. on "Robotics and Automation", Scottsdale, USA, pp.1334-1339.
5. Mittal, S., Tasch, U. and Wang, Y. (1993) A Redundant Actuation Scheme for Independent Modulations of Stiffness and Position of a Robotic Joint: Design, Implementation and Experimental Evaluation, DSC-Vol.49, *Advances in Robotics, Mechatronics, and Haptic Interfaces*, ASME, pp.247-256.
6. Cho, W., Tessar, D. and Freeman R. (1989) The Dynamic and Stiffness Modelling of General Robotic Manipulator Systems with Antagonistic Actuation, Proc. of the Int. Conf. "Robotics and Automation", Scottsdale, Arizona, pp.1380-1387.
7. Kostadinov, K. (1993) Accommodation control in the drive dynamic accuracy for positioning robot, 38 Internationales wissenschaftliches kolloquium, Tagungsband, ss. 100-108, Ilmenau
8. Kostadinov, K.Gr. and Parushev, P.R, (1987) Method of motion transition, Bulg. Patent No.44365.
9. Tadokoro, S. (1994) Control of Parallel Mechanisms, J. *Advanced Robotics*, Vol. 8, No. 6, pp. 559-571
10. Kostadinov, K., (1996) Impedance Controlled Actuators for Positioning Robots, Proc. of 3rd International Conference on Mechatronics and Machine Vision in Practice, Guimaraes-Portugal, vol.2, pp.243-248.
11. Kostadinov, K. and Boiadjiev, G. (1997) Dynamic Modelling of Impedance Controlled Actuators for Positioning Robots, In "Solid Mechanics and Its Applications" - Series Editor G.M.L. Gladwell vol.52, Kluwer Academic Publishers, pp.183-190.
12. Kostadinov, K., Vassillev, B. and Ranchev, S. (1996) Motion Control Study of Harmonic Actuator with Impedance Control at Some Impacts, in Proc. of the 6th Int. Symposium on Measurement and Control in Robotics, Brussels, pp.177-182.
13. Boiadjiev, G. (1996) Kinematic Sensible Directions of Manipulating Systems, in Proc. of the 6th Int. Symp. on Measurement and Control in Robotics, Brussels, pp.77-82.
14. Boiadjiev, G. and Kostadinov, K. (1997) Coefficients and Directions of the Dynamic Sensibility of Robot Manipulators, in Proc. of the 5ht Int. Workshop on Advanced Robotics & Intelligent Machines, Ed. J. Gray & D. Caldwell, P.7, Manchester, UK
15. Cheah, C.-C. and Wang, D. (1998) Learning Impedance Control for Robotic Manipulators, IEEE Trans. On *Robotics and Automation*, Vol.14, No.3, pp.452-465.
16. Caccavale, F., Natale, C., Sicoliano, B. and Villani, L. (1999) Six-DOF Impedance Control Based on Angle/Axis Representation, IEEE Trans. On *Robotics and Automation*, Vol.15, No.2, pp.289-300.

OPTIMIZING MULTIBODY SYSTEMS: SOME IMPLEMENTATIONS AND RESULTS*

R. KOVALEV
Mech.-Eng.
Bryansk State Technical University
bulv. 50-let Oktyabrya, 7, Bryansk, 241035, Russia
kovalyov@bitmcnit.bryansk.su

Abstract. Optimization of multibody systems is presented as a multicriteria optimization problem. The problem of forming goal function is still open due to wide variety of conflicting criteria, which as a rule has to be reduced to a scalar function. Implementation of an approach for optimizing dynamic systems based on the analytic hierarchy process for getting the scalar goal function is considered. The developed approach is implemented in program package for simulation of multibody system dynamics.

1. Introduction

The development of computing facilities, formalisms for automatic generation of equations of motion and numerical methods let increase the effectiveness of program packages for simulation of multibody dynamics thus much that it made possible the development of program packages for optimization of multibody system.

Design variables and performance criteria have to be defined for optimization problem solving. Such parameters of multibody system as inertia and geometrical data, stiffness and damping coefficients might be chosen as design variables. As it mentioned in [1, 2, 3] applications to technical problems clearly show that as a rule several conflicting technical specifications and goals have to be taken into consideration. Since there are several criteria the optimization problem has to be considered as a multicriteria. Due to some disadvantages of multicriteria optimization method strategies, which reduce the vector optimization problem to nonlinear programming problems are usually used [2]. There are quite a few methods for such reducing based on scalarization and hierarchization principles or a combination of them. The implementation of one of such methods, so called the analytic hierarchy process, is considered below.

2. Formulation of the optimization problem

Computer aided optimization of mechanical systems should be based on mathematical models. The multibody system approach gives us good representation of the system if we can neglect small deformations of its parts [1]. A multibody system consists of rigid

* Supported by RFBR under the grant 02-01-00364 and by scientific program «Universities of Russia – Basic Research»

bodies and ideal joints. A body may degenerate to a particle or to a body without inertia. The ideal joints include the rigid joints, the joints with completely given motion (rheonomic constraint) and the vanishing joints (free motion) [3].

Multibody system dynamics has been developing already for several decades [4]. Computer programs have been developed for automatic generation of equations of motion and its numerical solution. The most time-consuming part in an optimization of multibody systems is an estimation of scalar or vector goal function, which involves a numerical solution of motion equations for some time interval $t^1 < t < t^2$. An optimization strategy takes, in fact, the smaller part of total time efforts.

It is necessary to distinguish a simulation problem and an optimization problem. The optimization problem does not need the most complex model, it needs the most appropriate one [2].

Mathematical models involve a parameterization. The dynamic behavior of the model is completely determined by parameters like the mass and moments of inertia of each body, geometrical dimensions, damping and stiffness coefficients.

Optimization criteria are usually based on dynamical performances obtained as results of numerical experiments. Usually it is such performances like accelerations (riding comfort), reaction forces (strength in joints), etc.

There might be a single optimization criterion, but generally there are several specifications, goals and restrictions, so the design problem has to be considered as a multicriteria optimization problem.

Generally, optimization of multibody systems takes place during the beginning of development of a new technical system. The optimal values of parameters, which are found with respect to dynamical performance, however, might not be appropriate due to another (technological, cost) reasons. Obviously it might be necessary to involve non-dynamical criterions into consideration.

3. Multicriteria optimization

The problem of optimizing dynamic systems with respect to several conflicting criteria does not have a single optimal solution [1]. Edgeworth-Pareto (EP-) optimal points can be found. EP-optimal solutions are not unique and different points are not comparable. The theory of multicriteria optimization has shown that the optimum depends on additional decisions of the designer. Therefore, not all multicriteria optimization strategies seem to be appropriate for dynamic system design. Strategies which reduce the vector optimization problem to non-linear programming problems have proven to be very efficient [2]. Several such strategies based on the principles of scalarization, hierarchization or a combination of them have been developed [1].

In the case of scalarization [2], the objective functions are combined to a new utility function $u(\mathbf{p})$, where $\mathbf{p}$ is the vector of design variables, which will be optimized instead of the vector criterion.

$$u(\mathbf{p}) = \sum_{i=1}^{n} w_i \frac{f_i(\mathbf{p})}{f_i^*}, \quad \sum_{i=1}^{n} w_i = 1,$$

where $w_i \in [0,1]$ are weighting coefficients and f_i^* are scaling factors. This well known

approach has some disadvantages. Firstly, designer establishes weighting coefficients directly that leads to an insufficient validity of weight coefficients. It is shown in [5] that people are inclined to shift weight coefficients to the ends of range. Secondly, utility function depends on the $f_i(\mathbf{p})/f_i^*$ ratio linearly whereas non-linear dependence may corresponds to the optimization goal better .

Let us consider an approach based on the analytic hierarchy process, which keeps advantages of hierarchization and removes disadvantages of scalarization.

4. The analytic hierarchy process

The analytic hierarchy process was developed by Saaty and in [5] detailed information is available.

The method is based on principle of hierarchization, when the main most common goal consists of several more detailed sub-goals, each sub-goal of the first level consists of corresponding sub-goals of level two and so on. Every sub-goal has only one upper goal. Different sub-goals affect the upper goal with the different weight.

Further, the analytic hierarchy process involves the method to determine the strength with which the various elements in one level influence the elements on the next higher level, so that we may compute the relative strength of the impacts of the elements of the lowest level on the overall objectives. The method can be described as follows. Given one goal, *e*, and its sub-goals of the next level lower, compare the sub-goals pairwise in their strength of influence on *e*. Insert the agreed upon numbers, reflecting the comparison, in a matrix and find the eigenvector with the largest eigenvalue. The eigenvector provides the priority ordering, and the eigenvalue is a measure of the consistency of the judgment [5].

To insert the agreed upon numbers the designer has to compare every pair of sub-goals and give an answer for the question "how stronger the influence of sub-goal *B* on the upper goal than the influence of sub-goal *C* on it", this number will be included in the (*B, C*) matrix element. If *B* and *C* equally important then the number is 1, if *B* is weakly more important than *C* then the number is 3 and so on up to number 9 when the *B* is absolutely more important than *C*.

5. Measuring performance

After describing hierarchy of goals the designer should determine the way to obtain the strength (priority, measure of membership) of each alternative relative to each element of hierarchy of goals on the lower level. There are several methods.

The first one is the pairwise comparison, described above. This methods is used for measuring non-dynamic performances of systems such as practical feasibility, estimated cost, etc (see Figure 2). In this case the designer uses results of scanning for pairwise comparisons. This is the most general way, but at the same time the most time-consuming method.

In the cases when criteria have numerical representation obtained from results of simulation, its transformation to dimensionless scale [0, 1] can be done with the help of

the methods of standards or membership function method.

The membership function method uses the membership functions to map simulation results into dimensionless scale [0, 1]. Some frequently used membership functions are given in Figure 1. Here the normalized performance estimation is laid off as abscissa and the strength of alternative relative to criterion is laid off as ordinate.

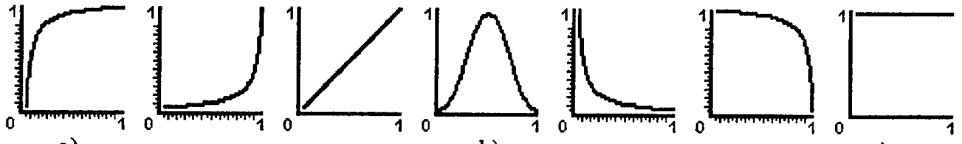


Figure 1. Frequently used membership functions

For example, the following comment may be given for the membership function in Figure 1a: "the less performance estimation the worse the strength of alternative, lower values are poorly acceptable"; for the Figure 1b function: "middle performance estimations are most acceptable".

Scale for membership functions abscissa may be defined in two forms: definite and indefinite. Definite scale is used when there is prior information about admissible or expected performance estimations. If any alternative has inadmissible performance estimation relative any criterion, it is considered as inadmissible due to that criterion and does not take part in the further comparison. If relative comparisons of performance estimations have no sense, but they all must be within a definite range then the membership function shown in Figure 1c may be used.

Indefinite scale is used when the designer has no prior information about the model behavior relative to a criterion. Then the minimal performance estimation corresponds to 0 abscissa value and the maximal one corresponds to 1. The rest performance estimations are distributed within this range proportionally.

The method of standards is the measuring relative to some standards. It is used when there are some standards, which can help us to classify the performances. For example, for vertical accelerations we can introduce tree levels: low, medium and high acceleration (see Figure 2). Based on results of numerical experiments, the program is able to refer alternatives to levels of accelerations.

6. Automated approach

Simulation of multibody systems is supported by program package "Universal Mechanism" (UM) and the special module for optimization and decision making is built in UM. At the first step the scanning of dynamic behavior of the optimized multibody system is fulfilled: variable parameters, parameter ranges and steps are defined, after that number of numerical experiments are executed in an automatic mode. It might take several hours and even several days. Then dynamic behavior of the system is

completely obtained. Then the designer has to make a decision which alternative (which point in a parameter space) is the best. It may be done by hand or with the help of the built-in module for decision making support.

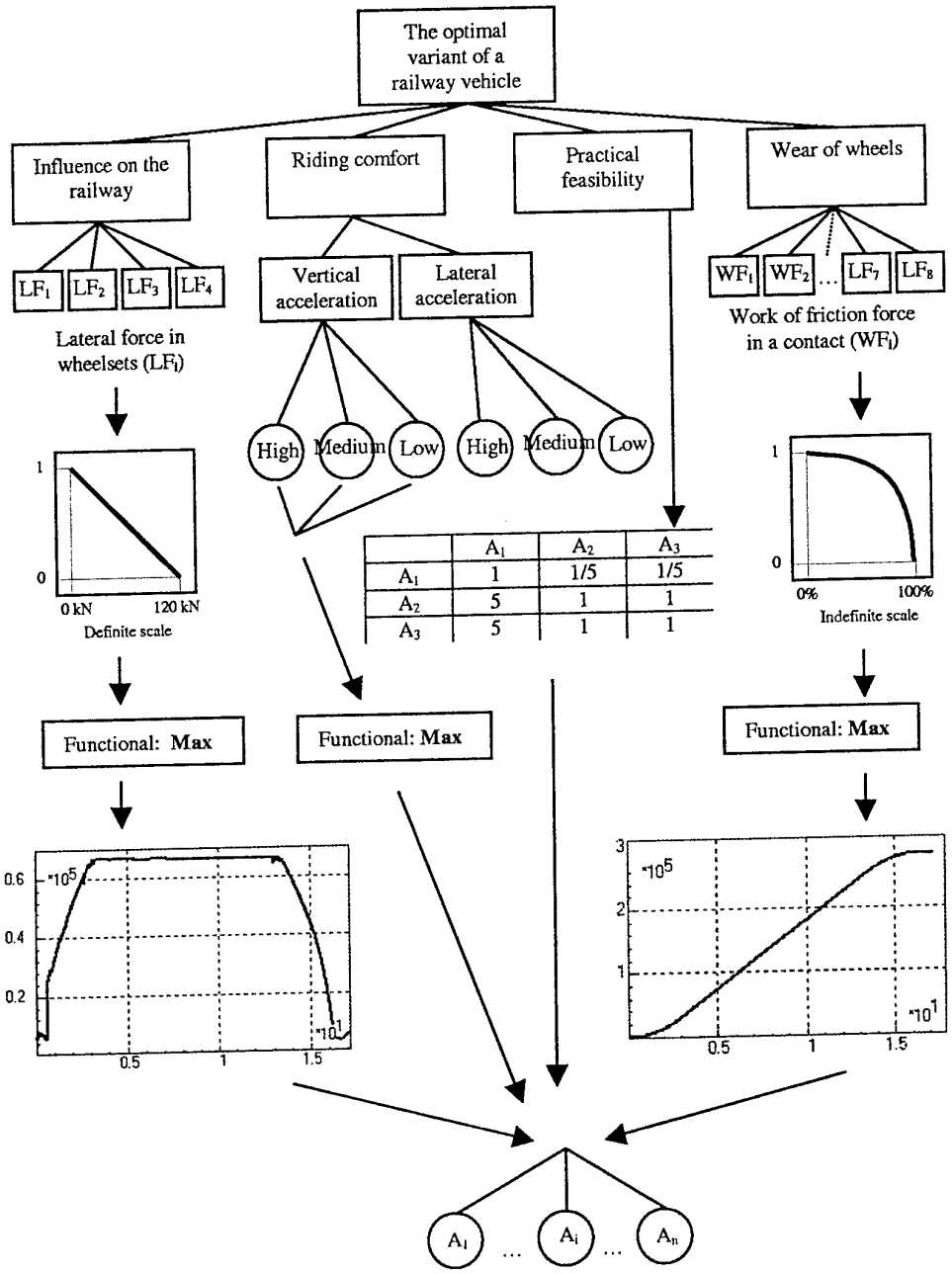


Figure 2. Hierarchy with the different types of measuring performance

In the latter case the designer should describe the hierarchy of goals and sub-goals (see Sect. 4, Figure 2). Then for each criterion on the last hierarchy level the measuring performance method is defined. For the pairwise comparison method (see. Sect. 4) the designer has to perform pairwise comparisons for every pair of alternative. For the methods of standards the designer defines the standards and assigns their ranges and chooses the appropriate functional (see Sect. 5). For the membership function method the designer chooses the membership function, define its scale (definite or indefinite) and choose the appropriate functional. In order to use a definite or indefinite scale or any described standard we have to transform the time history of a dynamical performance to a digit. It can be done with the help of one of available functionals: maximum, minimum, mean, root mean square, etc.

Further, the strength of each alternative relative to every criterion is automatically calculated and results are available. Information about the priority of alternatives relative to any goal (including the main one) is available as well.

References

1. Bestle, D., Eberhard, P. (1994) Automated Approach for Optimizing Dynamic Systems, *International Series of Numerical Mathematics*, Vol. 115, 225-235.
2. Bestle, D., Eberhard, P. (1996) Multi-criteria multi-model design optimization, in D. Bestle and W. Schielen (eds.), *IUTAM Symposium on Optimization of Mechanical Systems*, Kluwer Academic Publisher, 33-40.
3. Schiehlen, W. (1997) Multibody System Dynamics: Roots and Perspectives, *Multibody System Dynamics*, 149-188.
4. Schiehlen, W. (ed.) (1990) *Multibody System Handbook*, Springer, Berlin.
5. Saaty, T. (1980) *The Analytic Hierarchy Process*, McGraw-Hill.

A Formulation for Flexible Multibody Systems with Mixed Cartesian and Relative Coordinates

Lars Kübler and Peter Eberhard
Institute of Applied Mechanics, University of Erlangen-Nuremberg, Egerlandstr. 5, 91058 Erlangen, Germany, [kuebler,eberhard]@ltm.uni-erlangen.de

1. Introduction

With regard to the selection of system coordinates the methods used in the dynamic analysis of multibody systems (MBS) can in general be divided into two main approaches (Shabana, 1998). In the first approach an expanded system of dependent coordinates, e. g. *Cartesian coordinates*, is used to describe the system configuration. In the second approach, a minimum number of *relative coordinates* is used that corresponds to the mechanical degree of freedom of the system (Schiehlen, 1986) and provides a minimum number of ordinary differential equations (ODE) for tree-like structures.

The idea of the method proposed in this paper is to combine the advantages of both approaches. Several subsystems that can be rigid bodies, flexible bodies or rigid multibody system substructures are assembled in the global model. Their motion is kinematically constrained by mechanical joints or kinematic drivers, both described by nonlinear algebraic constraint equations. The motion of the elements within the substructures is described by using relative coordinates, while the relative motion of the subsystems is described in Cartesian space.

2. Choice of Coordinates for Multibody Systems

Two main approaches for the modeling of multibody systems, *relative* or *Cartesian coordinates*, respectively, are described in Sections 2.1 and 2.2. Both formulations are combined in Section 2.3 in order to share their advantages and overcome specific disadvantages of each method.

2.1. MBS Description with Relative Coordinates

The multibody system approach with relative coordinates is described in detail in (Schiehlen, 1986). Relative coordinates lead for open-chain configurations to a minimum number of ordinary differential equations,

using a set of independent generalized coordinates $\boldsymbol{y} \in \mathbb{R}^f$ corresponding to the degree of freedom f of the MBS. This is a main advantage in comparison to the approach with Cartesian coordinates where differential algebraic equations (DAE) of often much higher dimension have to be solved.

A spatial MBS consisting of n_b bodies with n_c holonomic constraints, and hence $f = 6n_b - n_c$ degrees of freedom, can be described, applying for example d'Alembert's principle, Hamilton's principle or the Newton-Euler formalism (Schiehlen, 1986) to the balances of linear and angular momentum. This yields the equations of motion

$$\boldsymbol{M}(t, \boldsymbol{y}) \cdot \ddot{\boldsymbol{y}} + \boldsymbol{k}(t, \boldsymbol{y}, \dot{\boldsymbol{y}}) = \boldsymbol{g}(t, \boldsymbol{y}, \dot{\boldsymbol{y}}) \tag{1}$$

with the symmetric, positive definite mass matrix $\boldsymbol{M} \in \mathbb{R}^{f \times f}$, the generalized centrifugal and Coriolis forces $\boldsymbol{k} \in \mathbb{R}^f$ and the generalized applied forces $\boldsymbol{g} \in \mathbb{R}^f$.

2.2. MBS Description with Cartesian Coordinates

The application of Cartesian coordinates has the advantage that the formulation of the equations of motion even for complex systems is straightforward. Beyond that the addition of new complex system components is often relatively easy. This can be, for example, very interesting when flexible bodies are added to the system.

A multibody system consisting of n_b interconnected rigid bodies, requires $6\,n_b$ coordinates in order to describe the system configuration in space, i. e., the positions $\boldsymbol{R}^i$ and orientations $\boldsymbol{S}^i$ of each body's reference frame. These coordinates, however, are not independent because of mechanical joints or kinematic drivers between adjacent bodies, described by a vector of n_c nonlinear kinematic constraint equations

$$\boldsymbol{c}(\boldsymbol{q}, t) = \boldsymbol{0}, \quad \boldsymbol{c} \in \mathbb{R}^{n_c} \tag{2}$$

where $\boldsymbol{q}$ is the vector of all Cartesian coordinates for all bodies.

The differential equations of motion of the system follow using Lagrange's equation or Hamilton's principle, see e.g. (Shabana, 1998), as

$$\boldsymbol{M} \cdot \ddot{\boldsymbol{q}} - \boldsymbol{C}_q^T \cdot \boldsymbol{\lambda} = \boldsymbol{Q}_e + \boldsymbol{Q}_v\,. \tag{3}$$

Here $\boldsymbol{M} \in \mathbb{R}^{6n_b \times 6n_b}$ is the mass matrix, $\boldsymbol{C}_q = \partial \boldsymbol{c}/\partial \boldsymbol{q} \in \mathbb{R}^{n_c \times 6n_b}$ is the constraint Jacobian matrix, $\boldsymbol{\lambda} \in \mathbb{R}^{n_c}$ is the vector of Lagrange multipliers, $\boldsymbol{Q}_e$ is the vector of externally applied forces and $\boldsymbol{Q}_v$ is a quadratic velocity vector that arises from differentiating the kinetic energy with respect to time and with respect to the generalized coordinates.

The differential equations (3) yield together with the algebraic equations (2), or respectively their mathematically equivalent first or second time derivatives, a DAE describing the global system.

2.3. Approach with mixed Relative and Cartesian Coordinates

The global system is assembled by n_s subsystems as illustrated in Figure 1. In correspondence to (1) the equations of motion of the different subsystems are given by

$$\boldsymbol{M}^i\left(t,\boldsymbol{y}^i\right)\cdot\ddot{\boldsymbol{y}}^i+\boldsymbol{k}^i\left(t,\boldsymbol{y}^i,\dot{\boldsymbol{y}}^i\right)=\boldsymbol{g}^i\left(t,\boldsymbol{y}^i,\dot{\boldsymbol{y}}^i\right),\quad i=1,2,\ldots,n_s\,. \tag{4}$$

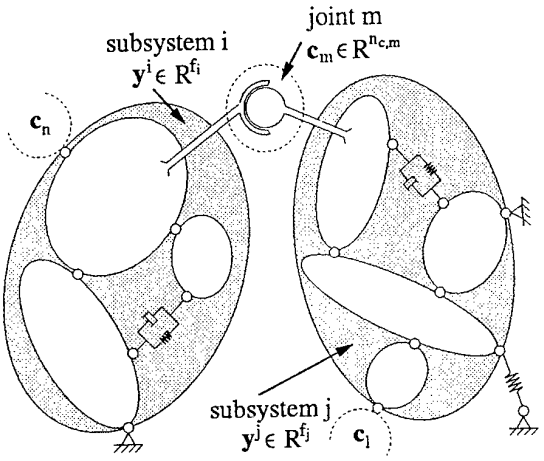


Figure 1. Assembly of subsystems forming the global multibody system

The subsystems are connected by n_j joints summerized in the global constraint vector $\boldsymbol{c}\in I\!R^{n_c}$. The generalized coordinates are assembled in the vector

$$\boldsymbol{q}=\left[\,\boldsymbol{y}^1\;\;\boldsymbol{y}^2\;\;\ldots\;\;\boldsymbol{y}^{n_s}\,\right]\in I\!R^{n_q}\quad\text{with}\quad n_q=\sum_{i=1}^{n_s}f^i\,. \tag{5}$$

The equations of motion of the global system can for example be derived by application of d'Alembert's principle, here given in the formulation of Lagrange (Eberhard, 2000)

$$\sum_{i=1}^{n_b}\left(\delta\boldsymbol{r}_i\cdot\left(m_i\,\boldsymbol{a}_i-\boldsymbol{f}_i^e\right)+\delta\boldsymbol{s}_i\cdot\left(\boldsymbol{I}_i\cdot\boldsymbol{\alpha}_i+\tilde{\boldsymbol{\omega}}_i\cdot\boldsymbol{I}_i\cdot\boldsymbol{\omega}_i-\boldsymbol{l}_i^e\right)\right)=0\,, \tag{6}$$

with the global number of bodies in the system n_b, the virtual displacements $\delta \boldsymbol{r}_i$ and rotations $\delta \boldsymbol{s}_i$, the externally applied forces $\boldsymbol{f}_i^e$ and torques $\boldsymbol{l}_i^e$ and the skew symmetric matrix $\tilde{\boldsymbol{\omega}}$ build by the components of the angular velocity vector $\boldsymbol{\omega} = [\omega_1, \omega_2, \omega_3]$.

Equation (6) can after some transformations be split up in terms for each subsystem. By application of kinematic relations and further transformations the local equations of motion of the subsystems can be identified

$$\sum_{j=1}^{n_s} \Bigg[\delta \boldsymbol{y}^j \cdot \Bigg(\underbrace{\sum_{k=1}^{n_b^j} \left(\boldsymbol{J}_{Tk}^{jT} \cdot m_k^j \boldsymbol{J}_{Tk}^j + \boldsymbol{J}_{Rk}^{jT} \cdot \boldsymbol{I}_k^j \cdot \boldsymbol{J}_{Rk}^j \right)}_{\boldsymbol{M}^j(t, \boldsymbol{y}^j)} \cdot \ddot{\boldsymbol{y}}^j + \underbrace{\sum_{k=1}^{n_b^j} \left(\boldsymbol{J}_{Tk}^{jT} \cdot m_k^j \bar{\boldsymbol{a}}_k^j + \boldsymbol{J}_{Rk}^{jT} \cdot \boldsymbol{I}_k^j \cdot \tilde{\bar{\boldsymbol{\alpha}}}_k^j + \boldsymbol{J}_{Rk}^{jT} \cdot \tilde{\boldsymbol{\omega}}_k^j \cdot \boldsymbol{I}_k^j \cdot \boldsymbol{\omega}_k^j \right)}_{\boldsymbol{k}^j(t, \boldsymbol{y}^j, \dot{\boldsymbol{y}}^j)} - \underbrace{\sum_{k=1}^{n_b^j} \left(\boldsymbol{J}_{Tk}^{jT} \cdot \boldsymbol{f}_k^{ej} + \boldsymbol{J}_{Rk}^{jT} \cdot \boldsymbol{l}_k^{ej} \right)}_{\boldsymbol{g}^j(t, \boldsymbol{y}^j, \dot{\boldsymbol{y}}^j)} \Bigg) \Bigg] = 0 \tag{7}$$

where n_b^j is the number of bodies in subsystem j and $\boldsymbol{J}_T^j$ and $\boldsymbol{J}_R^j$ are the Jacobians of translation and rotation.

By introduction of the global coordinates vector $\boldsymbol{q}$ from (5) and its variation, equation (7) can be summarized

$$\delta \boldsymbol{q} \cdot (\boldsymbol{M} \cdot \ddot{\boldsymbol{q}} + \boldsymbol{k} - \boldsymbol{g}) = \boldsymbol{0} \quad \forall \delta \boldsymbol{q} : \boldsymbol{C}_q^T \cdot \delta \boldsymbol{q} = \boldsymbol{0} \tag{8}$$

with the global constraint Jacobian matrix

$$\boldsymbol{C}_q = \frac{\partial \boldsymbol{c}}{\partial \boldsymbol{q}} = \left[\left(\frac{\partial \boldsymbol{c}^1}{\partial \boldsymbol{q}} \right)^T \left(\frac{\partial \boldsymbol{c}^2}{\partial \boldsymbol{q}} \right)^T \dots \left(\frac{\partial \boldsymbol{c}^{n_j}}{\partial \boldsymbol{q}} \right)^T \right]^T \in I\!R^{n_c \times n_q}. \tag{9}$$

In (8) only $f_g = n_q - n_c$ variations of $\delta \boldsymbol{q}$ are independent. By introducing a set of Lagrange multipliers $\boldsymbol{\lambda} \in I\!R^{n_c}$ as in Section 2.2, (8) can be written in a way valid for arbitrary variations of $\boldsymbol{q}$. This gives the equations of motion of the global multibody system

$$\boldsymbol{M}(t, \boldsymbol{q}) \cdot \ddot{\boldsymbol{q}} + \boldsymbol{k}(t, \boldsymbol{q}, \dot{\boldsymbol{q}}) - \boldsymbol{C}_q^T(t, \boldsymbol{q}) \cdot \boldsymbol{\lambda} = \boldsymbol{g}(t, \boldsymbol{q}, \dot{\boldsymbol{q}}) \tag{10}$$

which form a DAE combined with (2), or the analytically equivalent first or second time derivatives of (2).

3. Constraint Formulation for the Connection of Substructures

Most of the practically used kinematic constraints can be built by setting algebraic relations between vectors defined on the bodies, as discussed in detail in (Nikravesh, 1988), (Shabana, 1998), or (Serban and Haug, 1988).

In terms of accuracy and stability of the numerical simulation an index 2 DAE formulation performs best (Arnold, 1998), requiring the partial derivatives $\partial \boldsymbol{c}/\partial \boldsymbol{q}$ and the time derivatives $\dot{\boldsymbol{c}}$. In order to determine these derivatives analytically it is particularly attractive to transform the orientation matrix at the connection points to unit quaternions, e. g. using the algorithm proposed in (Shoemake, 1994). Unit quaternions allow for a proper derivation of mathematical relations for observed coordinate frames. Many basic identities and analytical relations are given, e. g. in (Nikravesh, 1988). They are described as follows

$$\boldsymbol{p} = [e_0\,,\, \boldsymbol{e}] \quad \text{with} \quad e_0 = \cos\frac{\Phi}{2}\,, \quad \boldsymbol{e} = \boldsymbol{u}\sin\frac{\Phi}{2}\,, \tag{11}$$

where Φ is the rotation angle about a axis described by the unit vector $\boldsymbol{u}$ with the additional constraint $\boldsymbol{u}\cdot\boldsymbol{u} = 1$ or rather $\boldsymbol{p}\cdot\boldsymbol{p} - 1 = 0$.

In their paper Serban and Haug (1988) already derived the required analytical derivatives for constraint vectors using unit quaternions. For the mixed approach presented in this paper additionally the dependencies of the unit quaternions on the relative coordinates describing the rigid body substructures have to be considered. The necessary relations are derived in the following section.

3.1. Derivatives of Quaternions with Respect to the Generalized Coordinates of the Global System

Without loss of generality, a global system consisting of two subsystems, connected at point P, is observed in this section with the generalized coordinates $\boldsymbol{q} = [\,\boldsymbol{y}_A\ \boldsymbol{y}_B\,]$, as illustrated in Figure 2. The rotation matrices at the observer frames $\boldsymbol{S}_{0A}(\boldsymbol{y}_A,\, t)$ and $\boldsymbol{S}_{0B}(\boldsymbol{y}_B,\, t)$ on P, described on body i and body j, are transformed to quaternions $\boldsymbol{p}_A(\boldsymbol{y}_A,\, t)$ and $\boldsymbol{p}_B(\boldsymbol{y}_B,\, t)$.

The time derivative of unit quaternions can be expressed by the identity (Nikravesh, 1988)

$$\dot{\boldsymbol{p}} = \frac{1}{2}\,\boldsymbol{G}^T\cdot\boldsymbol{\omega} \tag{12}$$

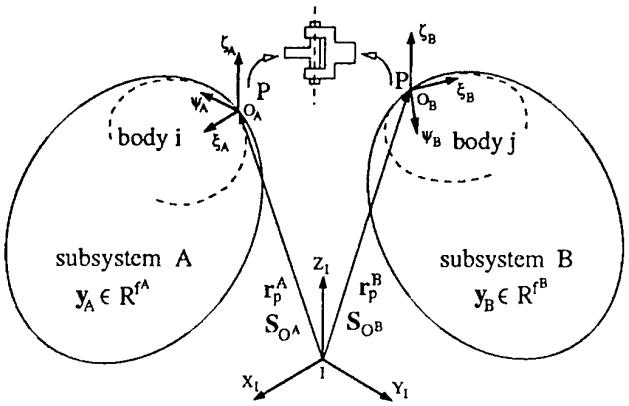


Figure 2. Schematic representation of the connection of two subsystems

with the vector of angular velocity $\boldsymbol{\omega}$ and the matrix $\boldsymbol{G} \in I\!R^{3\times 4}$ defined by Nikravesh as

$$\boldsymbol{G} := \begin{bmatrix} -e_1 & e_0 & -e_3 & e_2 \\ -e_2 & e_3 & e_0 & -e_1 \\ -e_3 & -e_2 & e_1 & e_0 \end{bmatrix} = [-\boldsymbol{e},\ \tilde{\boldsymbol{e}} + e_0 \boldsymbol{I}] \,. \tag{13}$$

Starting from equation (12) the partial derivatives $\partial \boldsymbol{p}_A / \partial \boldsymbol{q}$ and $\partial \boldsymbol{p}_B / \partial \boldsymbol{q}$ can be determined for both subsystems. The procedure is exemplary given here for $\partial \boldsymbol{p}_A / \partial \boldsymbol{q}$. It follows from (12)

$$\dot{\boldsymbol{p}}_A = \frac{1}{2} \boldsymbol{G}_A^T \cdot (\boldsymbol{J}_{RA}^i \cdot \dot{\boldsymbol{y}}_A + \bar{\boldsymbol{\omega}}_A^i) \tag{14}$$

where $\boldsymbol{J}_{RA}^i$ is the Jacobian matrix of rotation of body i in subsystem A and $\bar{\boldsymbol{\omega}}_A^i$ is the local angular velocity of body i. Since we have the dependencies $\boldsymbol{p}_A = \boldsymbol{p}_A(\boldsymbol{y}_A, t)$ the vector $\dot{\boldsymbol{p}}_A$ can also be expressed as

$$\dot{\boldsymbol{p}}_A = \frac{\partial \boldsymbol{p}_A}{\partial \boldsymbol{y}_A} \cdot \dot{\boldsymbol{y}}_A + \frac{\partial \boldsymbol{p}_A}{\partial t} \,. \tag{15}$$

By comparison of the coefficients in equations (14) and (15) it follows

$$\frac{\partial \boldsymbol{p}_A}{\partial \boldsymbol{y}_A} = \frac{1}{2} \boldsymbol{G}_A^T \cdot \boldsymbol{J}_{RA}^i \,, \tag{16}$$

and hence the partial derivatives with respect to the global generalized coordinates $\boldsymbol{q} = [\boldsymbol{y}_A\ \boldsymbol{y}_B]$ can written as

$$\frac{\partial \boldsymbol{p}_A}{\partial \boldsymbol{q}} = \left[\frac{\partial \boldsymbol{p}_A}{\partial \boldsymbol{y}_A} \mid \frac{\partial \boldsymbol{p}_A}{\partial \boldsymbol{y}_B} \right] = \left[\frac{1}{2} \boldsymbol{G}_A^T \cdot \boldsymbol{J}_{RA}^i \mid \boldsymbol{0} \right] \tag{17}$$

$$\frac{\partial \boldsymbol{p}_B}{\partial \boldsymbol{q}} = \left[\frac{\partial \boldsymbol{p}_B}{\partial \boldsymbol{y}_A} \mid \frac{\partial \boldsymbol{p}_B}{\partial \boldsymbol{y}_B} \right] = \left[\boldsymbol{0} \mid \frac{1}{2} \boldsymbol{G}_B^T \cdot \boldsymbol{J}_{RB}^j \right] \tag{18}$$

where (18) can be found utilizing the procedure above for subsystem B.

4. Example: Spatial Slider–Crank Mechanism

In this chapter the proposed approach with mixed Cartesian and relative coordinates is verified by application to a spatial slider-crank mechanism, similar to an example in (Serban and Haug, 1988), see Figures 3 and 4.

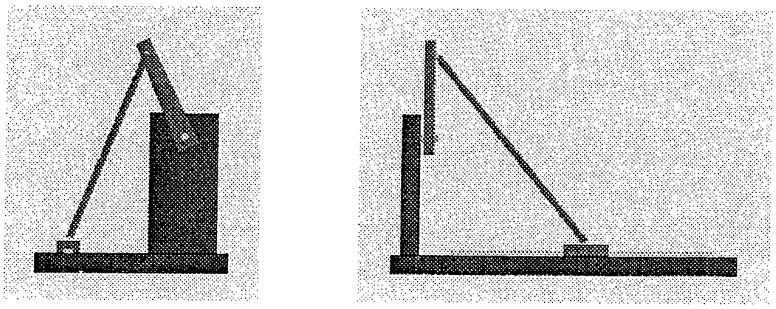


Figure 3. Spatial slider-crank mechanism

First the slider-crank mechanism is studied using a description with pure Cartesian coordinates. In order to verify the performance of the mixed approach, in a second step the mechanism is modeled using a subsystem described with NEWEUL (Kreuzer, 1991) in relative coordinates. The subsystem is joined to the inertial body with the loop closing translational joint. This leads to a DAE with 7 ODE and 5 algebraic equations, in comparison to the previous model with 18 ODE and 16 algebraic equations.

In Table I the computation times are compared for the simulation time of 20 seconds for both models. It can be seen that even for this relatively simple test example the efficiency can be strongly increased using the mixed approach.

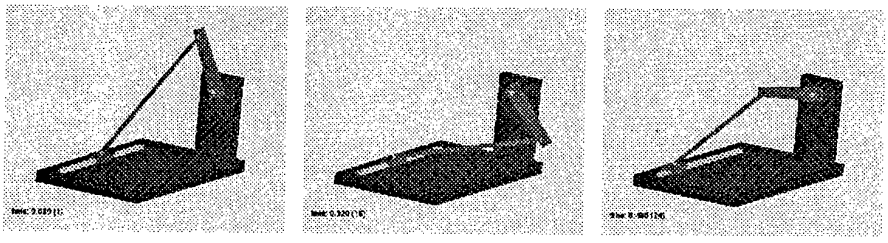


Figure 4. Frames from the animation of the slider-crank mechanism

Table I. Comparison of computation times

model	simulation time	computation time
Cartesian approach	$20s$	$66.2s$
mixed approach	$20s$	$14.7s$

5. Conclusions and Outlook

A mixed coordinate method which combines the efficiency of the relative coordinate approach with the generality of the formulation with Cartesian coordinates was presented. In order to determine the necessary derivatives of the constraint vector a transformation to unit quaternions was carried out for the reference coordinate frames. In addition to analytical derivatives found in literature, the description of subsystems in relative coordinates demanded for further derivatives, relating the unit quaternions and the generalized coordinates.

In order to verify the performance of the mixed approach, computation times were compared for two different models of a slider-crank mechanism. It was found that the efficiency strongly increased using the presented approach.

In the next step of implementation also flexible bodies will be attached to the systems. Thereby, the implemented description of the subsystems will be fully utilized.

References

M. Arnold. *Zur Theorie und zur Lösung von Anfangswertproblemen für differentiell-algebraische Systeme von höherem Index.* VDI–Fortschritt-Berichte, Series 20, No. 264, Düsseldorf: VDI-Verlag, 1998.

P. Eberhard. *Kontaktuntersuchungen durch hybride Mehrkörpersystem / Finite Elemente Simulationen.* Aachen: Shaker Verlag, 2000.

E. Kreuzer and G. Leister. *Programmsystem NEWEUL'90.* Manual AN–24. Institute B of Mechanics, University of Stuttgart, 1991.

P. Nikravesh. *Computer–Aided Analysis of Mechanical Systems.* Englewood Cliffs: Prentice Hall, 1988.

W. Schiehlen. *Technische Dynamik.* Stuttgart: B.G. Teubner, 1986.

R. Serban and E. J. Haug. Analytical derivatives for multibody system analysis. *Mechanics of Structures and Machines*, 26, pp. 145–173, 1988.

A. Shabana. *Dynamics of Multibody Systems.* Cambridge: University Press, 1998.

K. Shoemake. *Quaternions.* Department of Computer and Information Science, University of Pennsylvania, online paper:
`http://www.daimi.au.dk/~mbl/cgcourse/wiki/_files/shoemake94.pdf`.

Forward dynamics of multibody mechanisms using an efficient algorithm based on canonical momenta

Dirk Lefeber*, Joris Naudet*, Zdravko Terze† and Frank Daerden*

Abstract

A new method for establishing the equations of motion of multibody mechanisms based on canonical momenta is introduced in this paper. In absence of constraints, the proposed forward dynamics formulation results in a Hamiltonian set of $2n$ first order ODE's in the generalized coordinates q and the canonical momenta p. These Hamiltonian equations are derived based on a recursive Newton-Euler formulation. As an example, it is shown how, in case of a serial structure with rotational joints, an $O(n)$ formulation is obtained. The amount of arithmetical operations is considerably less than acceleration based $O(n)$ formulations.

1 Introduction

A lot of research has been done during the last decades to find new algorithms, new numerical integration techniques and better implementation methods to speed up the calculation of the motion of complex multibody mechanisms. Amongst many others, Featherstone [4], Kane and Levinson [7], Rosenthal [9] and Vukobratović [10] put significant efforts in finding efficient order N methods to derive the equations of motion. Bayo and Avello [2] developed techniques to integrate these equations in a stable and efficient way. Work has also been done to implement algorithms on a parallel computing architecture (Bae et al. [1]). All this research and the fast evolution of computer technology resulted in quite fast simulations nowadays. These simulations, however, involve mechanisms of ever increasing complexity (large amount of parts, elasticity, friction, backlash) and demand an ever increasing accuracy and, hence, number of computations. It is therefore interesting to continue this research in order to find more efficient algorithms. This article takes a step in that direction and presents a new, canonical momenta based algorithm, which allows a speedup of simulations by reducing the number of operations required to obtain the equations of motion. Nearly all efficient algorithms, whether they are based on the Newton-Euler equations, the Lagrange formulation or the principle of virtual work or virtual power, involve the computation of accelerations. This implies calculating the Coriolis and centrifugal forces and the solution of the forward kinematics. The canonical momenta based algorithm, however, is derived from a special form of the Newton-Euler equations and results in a formulation without accelerations, namely a set of Hamiltonian equations. Therefore, the number of arithmetical operations is strongly reduced. A few simplifications are made for the sake of clarity. Only serial structures with perfect revolute joints of one degree of freedom are considered. And, as usual, rigid bodies and a fixed base is assumed. The case of a floating base can easily be derived. An effort is made to explain the essence and details of the algorithm by situating it in the theory of Hamilton. The next section is therefore entirely dedicated to a review of Hamilton's equations. Then a special form of the Newton-Euler equations allowing a more logical construction of the algorithm is introduced. In section 4, the first set of Hamilton's equations is derived. In the subsequent section, the second set is found. Conclusions are drawn in the last section.

*Free University of Brussels, Department of Mechanical Engineering
†University of Zagreb, Department of Aerospace Engineering

2 Hamilton's equations

Mechanical systems are governed by the principle of least action, which can be formulated by means of the well-known Hamiltonian equations (see Goldstein [6]):

$$\dot{\mathbf{q}} = \frac{\partial H}{\partial \mathbf{p}} \tag{1a}$$

$$\dot{\mathbf{p}} = -\frac{\partial H}{\partial \mathbf{q}} + \mathbf{Q} - \Phi_{\mathbf{q}}^T \boldsymbol{\lambda} \tag{1b}$$

$$\boldsymbol{\Phi}(\mathbf{q}, t) = 0 \tag{1c}$$

This is a system of $2n$ first order differential equations and l kinematic constraint equations. It is called a set of mixed differential algebraic equations (DAE). H is the Hamiltonian function, $\mathbf{q}$ are the generalized coordinates. The vector $\mathbf{p}$ represent the so-called canonical momenta, extensions of the concept of linear and angular momenta to generalized coordinates. These canonical momenta are defined as:

$$\mathbf{p} = \frac{\partial L}{\partial \dot{\mathbf{q}}} \tag{2}$$

with L the Langrangian function. Functions $\boldsymbol{\Phi}$ are the kinematic constraint equations. Vector $\mathbf{Q}$ stands for the known generalized external forces, $\Phi_{\mathbf{q}}$ is the Jacobian matrix and $\boldsymbol{\lambda}$ represents the Lagrange parameters. DAE's are characterized by a so-called differential index. The acceleration based formulations have an index of 3, the Hamiltonian formulation has index 2 [5]. As shown by Brenan et al. [3], index 2 DAE's have a better behavior during numerical integration. Hence, the use of canonical momenta may be numerically advantageous compared to the use of accelerations. In the case of serial structures, and using the joint coordinates as generalized coordinates, no constraint equations (1c) are needed and the last term of (1b) disappears. Rewriting these equations in a more general form will ease future comparisons when motivating some steps in the algorithm:

$$\dot{\mathbf{q}} = F(\mathbf{q}, \mathbf{p}, t) \tag{3a}$$

$$\dot{\mathbf{p}} = G(\mathbf{q}, \mathbf{p}, t) \tag{3b}$$

Hamiltonian equations are computationally intensive to derive straightforwardly, for the Hamilton function H has to be established from the Lagrangian function L which already requires a lot of arithmetical operations. This is probably the reason for the lack of interest in Hamilton's equations in the domain of multibody mechanics. In acceleration based $O(n)$ algorithms, the equations of motion are found by recursion. This way the direct derivation of the Lagrangian *function* L is avoided and much faster evaluations of the equations of motion are obtained. However, it also seems possible to find an $O(n)$ algorithm based on canonical momenta. That algorithm, as will be shown in the following sections, has a reduced number of operations, compared even to the most efficient acceleration based algorithms. This advantage and the improved numerical behavior makes it a very promising alternative.

3 Newton-Euler in relative axes

The classical formulation of the Newton-Euler equations for a single free moving body is given by

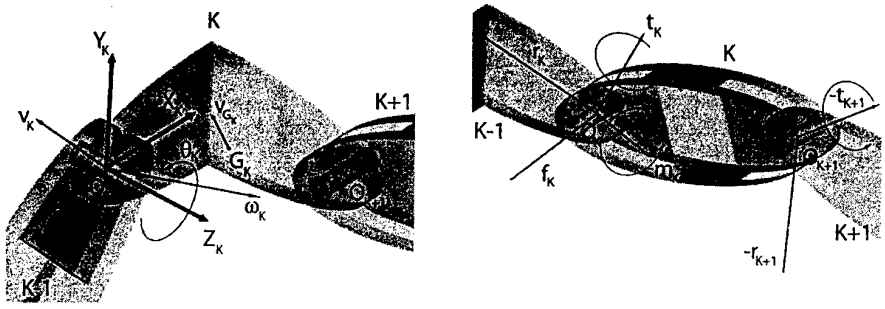
$$m\frac{d^0 \mathbf{v}_G}{dt} = \mathbf{f} \tag{4a}$$

$$J_G\frac{d^K \boldsymbol{\omega}}{dt} + \boldsymbol{\omega} \times J_G\boldsymbol{\omega} = \mathbf{m}_G \tag{4b}$$

The first equation is typically written in an inertial reference frame (notation $\frac{d^0}{dt}$), while the second is formulated in a frame K fixed to the body ($\frac{d^K}{dt}$). The force and the torque that act on the object are represented by $\mathbf{f}$ and $\mathbf{m}$. The index G denotes that the momenta are taken with respect to the center of mass. The matrix J is the inertia tensor, m is the mass of the body and $\boldsymbol{\omega}$ the angular

velocity refered to the inertial axes. Instead of trying to find an algorithm directly starting from the Hamiltonian equations, the Newton-Euler equations (4) are reformulated in relative axes, and written with respect to a point on the joint axis (the local z-axis). Note that the derivative to time of a free vector $\mathbf{x}$ in a rotating frame is given by:

$$\frac{d^0\mathbf{x}}{dt} = \frac{d^K\mathbf{x}}{dt} + \boldsymbol{\omega} \times \mathbf{x} \tag{5}$$



(a) Kinematics (b) Dynamics

Figure 1: Notation on body K

After some mathematical manipulations, equations (4) can be written as:

$$\begin{pmatrix} m & m\widetilde{GO} \\ m\widetilde{OG} & J \end{pmatrix}_K \begin{pmatrix} \dot{\mathbf{v}} \\ \dot{\omega} \end{pmatrix}_K + \begin{pmatrix} \tilde{\omega} & 0 \\ 0 & \tilde{\omega} \end{pmatrix}_K \begin{pmatrix} m & m\widetilde{GO} \\ m\widetilde{OG} & J \end{pmatrix}_K \begin{pmatrix} \mathbf{v} \\ \omega \end{pmatrix}_K = \begin{pmatrix} \mathbf{f} \\ \mathbf{m} + m\mathbf{v}_G \times \mathbf{v} \end{pmatrix}_K$$

The index K denotes which body of the mechanism is refered to. By convention, all momenta are then taken with respect to the point O_K on the joint axis of the body (see figure 1(a)). $\dot{\mathbf{x}}$ stands for the time derivative in *local* axes, e.g. $\dot{\omega}_K = \frac{d^K\omega_K}{dt}$. $\tilde{x}$ is a skew-symmetric matrix constructed from the vector $\mathbf{x}$ and is an alternative notation for the cross product.
The 6-dimensional momentum *vector* is defined as follows:

$$\mathbf{P}_K = \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K = \begin{pmatrix} m & m\widetilde{GO} \\ m\widetilde{OG} & J \end{pmatrix}_K \begin{pmatrix} \mathbf{v} \\ \omega \end{pmatrix}_K = M_K \begin{pmatrix} \mathbf{v} \\ \omega \end{pmatrix}_K \tag{6}$$

This is *not* the same vector as was used in the previous section to denote the canonical momenta $\mathbf{p}$. Inspection of $\mathbf{P}$ reveals that it is nothing more than a concatenation of the linear ($\mathbf{p}_l$) and angular ($\mathbf{p}_a$) momenta of the rigid body. M is called the mass matrix. Substitution of vector $\mathbf{P}$ in the equations of motion and observing that $\mathbf{p}_l = m\mathbf{v}_G$ results in the following concise formulation:

$$\begin{pmatrix} \dot{\mathbf{p}}_l \\ \dot{\mathbf{p}}_a \end{pmatrix}_K + \begin{pmatrix} \tilde{\omega} & 0 \\ \tilde{v} & \tilde{\omega} \end{pmatrix}_K \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K = \begin{pmatrix} \mathbf{f} \\ \mathbf{m} \end{pmatrix}_K \tag{7}$$

This expression can be written, since $\dot{M}_K = 0$.

4 First set of equations (G)

In this section we will derive one set of equations depicted in (3), namely the one involving the evaluation of the function G. The other set will be discussed in the next section. Rewriting (7) for the last body N and splitting the external forces and torques in the known parts $\mathbf{f}$ and $\mathbf{m}$ and the unknown parts $\mathbf{r}$ and $\mathbf{t}$ —resulting from the interaction with the previous body $N-1$— gives

$$\begin{pmatrix} \dot{\mathbf{p}}_l \\ \dot{\mathbf{p}}_a \end{pmatrix}_N + \begin{pmatrix} \tilde{\omega} & 0 \\ \tilde{v} & \tilde{\omega} \end{pmatrix}_N \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_N = \begin{pmatrix} \mathbf{f}+\mathbf{r} \\ \mathbf{m}+\mathbf{t} \end{pmatrix}_N \tag{8}$$

According to the assumptions made in section 1, each body introduces one degree of freedom q. Hence, to describe the motion of the mechanism only one set of two first order differential

equations is required for each body. Taking the joint angles θ to be the generalized coordinates is an obvious choice. The direction of the relative movement is defined by the unit vector $\mathbf{e}_z$ along the local z-axis. There is no reaction torque in that direction. Thus, the equation resulting from the projection of the angular part of (8) on $\mathbf{e}_z$ can be computed, provided the linear and angular velocities are known. The issue of the unknown velocities will be considered in the next section. For the remaining bodies, the equations become more involved, as there are two locations where reactions occur. Body $K = N - 1$ is connected with bodies $K - 1$ and $K + 1$. Therefore the equations of motion can be written as

$$\begin{pmatrix} \dot{\mathbf{p}}_l \\ \dot{\mathbf{p}}_a \end{pmatrix}_K + \begin{pmatrix} \tilde{\omega} & 0 \\ \tilde{v} & \tilde{\omega} \end{pmatrix}_K \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K = \begin{pmatrix} \mathbf{f}+\mathbf{r} \\ \mathbf{m}+\mathbf{t} \end{pmatrix}_K - {}^K\mathcal{T}_N^F \begin{pmatrix} \mathbf{r} \\ \mathbf{t} \end{pmatrix}_N \tag{9}$$

Now, besides the reactions resulting from body $K - 1$, additional reactions $-\mathbf{r}_{K+1}$ and $-\mathbf{t}_{K+1}$ from body $K + 1$ act on K (Fig. 1(b)). As we saw, the projection on the z-axis of reaction torque $\mathbf{t}_K$ is zero, but the reaction force $-\mathbf{r}_{K+1}$ is generally not directed towards joint point O_K and will therefore produce a torque about the local z-axis. This makes things more complicated. By convention, the reactions from body N are taken with respect to point O_N on the joint axis. To transmit these reactions to body $N - 1$, the transformation matrix ${}^K\mathcal{T}_N^F$ is used:

$${}^K\mathcal{T}_N^F = \begin{pmatrix} I & 0 \\ \widetilde{O_K O_N} & I \end{pmatrix} \tag{10}$$

Note that this matrix is constant in the local reference frame. Observe also that the velocities transform in a similar way:

$$\begin{pmatrix} \mathbf{v}_N \\ \omega_K \end{pmatrix} = {}^N\mathcal{T}_K^V \begin{pmatrix} \mathbf{v}_K \\ \omega_K \end{pmatrix} = \begin{pmatrix} I & \widetilde{O_N O_K} \\ 0 & I \end{pmatrix} \begin{pmatrix} \mathbf{v}_K \\ \omega_K \end{pmatrix} \tag{11}$$

The relationship between both transformation matrices is given by:

$${}^K\mathcal{T}_N^F = ({}^N\mathcal{T}_K^V)^T \tag{12}$$

The additional reactions along the z-axis cannot be ignored, as in the previous section, but can be eliminated by means of the equations of motion for body N (8). Grouping the similar terms and remembering to derive with respect to the correct coordinate system using (5) gives

$$\begin{aligned} \frac{d^K}{dt}\left[\begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K + {}^K\mathcal{T}_N^F \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_N\right] + \begin{pmatrix} \tilde{\omega} & 0 \\ \tilde{v} & \tilde{\omega} \end{pmatrix}_K \left[\begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K + {}^K\mathcal{T}_N^F \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_N\right] \\ = \begin{pmatrix} \mathbf{f}+\mathbf{r} \\ \mathbf{m}+\mathbf{t} \end{pmatrix}_K + {}^K\mathcal{T}_N^F \begin{pmatrix} \mathbf{f} \\ \mathbf{m} \end{pmatrix}_N \end{aligned} \tag{13}$$

After defining the *articulated* momentum vectors $\mathbf{p}_{l,a}^*$ and the *accumulated* forces and torques $\mathbf{f}^*$ and $\mathbf{m}^*$ as

$$\begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K^* = \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K + {}^K\mathcal{T}_N^F \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_N^* \tag{14}$$

$$\begin{pmatrix} \mathbf{f} \\ \mathbf{m} \end{pmatrix}_K^* = \begin{pmatrix} \mathbf{f} \\ \mathbf{m} \end{pmatrix}_K + {}^K\mathcal{T}_N^F \begin{pmatrix} \mathbf{f} \\ \mathbf{m} \end{pmatrix}_N^* \tag{15}$$

a concise system of equations is obtained with the same appearance as (8):

$$\begin{pmatrix} \dot{\mathbf{p}}_l \\ \dot{\mathbf{p}}_a \end{pmatrix}_K^* + \begin{pmatrix} \tilde{\omega} & 0 \\ \tilde{v} & \tilde{\omega} \end{pmatrix}_K \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_K^* = \begin{pmatrix} \mathbf{f}^*+\mathbf{r} \\ \mathbf{m}^*+\mathbf{t} \end{pmatrix}_K \tag{16}$$

Here again, the projection of the angular part on the joint axis $\mathbf{e}_z$ leads to one of the Hamiltonian equations. It can be proved that the element obtained by the same projection of the articulated

momentum vector $\mathbf{p}^*_{a_K}$ is the canonical momentum conjugated to θ_K. Therefore, this projection of the articulated momentum vector will be denoted with p_K, in accordance with the notation in section 2. The proof is based on the construction of the Langrangian function L and its partial derivatives (2). In summary, the function G has been identified, but can only be evaluated with known values of the velocities. These will be derived in next section.

5 Second set of equations (F)

To obtain the second set of Hamiltonian equations, the one involving the function evaluation F (3a), the joint velocities vector $\dot{\boldsymbol{\theta}}$ need to be expressed as a function of the canonical momenta vector $\mathbf{p}$ and the joint angles vector $\boldsymbol{\theta}$. This can be done starting from the expression of $\mathbf{P}_N$ in terms of the linear and angular velocities $\mathbf{v}_N$ and $\boldsymbol{\omega}_N$ (6) and writing the angular velocity as an explicit function of $\dot{\theta}_N$.

$$\boldsymbol{\omega}_N = \boldsymbol{\omega}_K + \dot{\theta}_N \mathbf{e}_{z_N} \tag{17}$$

Substitution in (6), projection on the angular z-axis and rearranging the terms gives an expression for the joint angle:

$$\dot{\theta}_N = \frac{1}{J_{zz_N}} \left[p_N - \begin{pmatrix} 0 & \mathbf{e}_z^T \end{pmatrix}_N M_N \begin{pmatrix} \mathbf{v}_N \\ \boldsymbol{\omega}_K \end{pmatrix} \right] \tag{18}$$

The scalar J_{zz} is a shorter notation for $\begin{pmatrix} 0 & \mathbf{e}_z^T \end{pmatrix} J \begin{pmatrix} 0 & \mathbf{e}_z \end{pmatrix}^T$. The expression is of the required form, as the Cartesian and angular velocities are functions of the joint velocities of all inboard links. If similar equations are found for all other bodies, the velocities can be computed recursively starting from the base. These equations can be obtained by first eliminating $\dot{\theta}_N$ from (6), by means of (17) and (18), and rearranging the terms:

$$\mathbf{P}_N = \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}_N = M'_N \begin{pmatrix} \mathbf{v}_N \\ \boldsymbol{\omega}_K \end{pmatrix} + \begin{pmatrix} \mathbf{d}_l \\ \mathbf{d}_a \end{pmatrix}'_N \tag{19}$$

with

$$M'_N = M_N - \frac{1}{J_{zz_N}} M_N \begin{pmatrix} 0 \\ \mathbf{e}_z \end{pmatrix}_N \begin{pmatrix} 0 & \mathbf{e}_z^T \end{pmatrix}_N M_N \quad \text{and} \quad \mathbf{D}'_N = \begin{pmatrix} \mathbf{d}_l \\ \mathbf{d}_a \end{pmatrix}'_N = \frac{1}{J_{zz_N}} p_N M_N \begin{pmatrix} 0 \\ \mathbf{e}_z \end{pmatrix}_N \tag{20}$$

M' is defined as the reduced mass matrix. $\mathbf{D}'$ is a remainder term. Substitution of (19) in (14) results in a desired formulation.

$$\mathbf{P}^*_K = \begin{pmatrix} \mathbf{p}_l \\ \mathbf{p}_a \end{pmatrix}^*_K = M^*_K \begin{pmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{pmatrix}_K + \begin{pmatrix} \mathbf{d}_l \\ \mathbf{d}_a \end{pmatrix}_K \tag{21}$$

with

$$M^*_K = M_K + {}^K\mathcal{T}^F_N M'_N {}^N\mathcal{T}^V_K \qquad \text{and} \qquad \mathbf{D}_K = \begin{pmatrix} \mathbf{d}_l \\ \mathbf{d}_a \end{pmatrix}_K = {}^K\mathcal{T}^F_N \begin{pmatrix} \mathbf{d}_l \\ \mathbf{d}_a \end{pmatrix}'_N \tag{22}$$

M^* is the articulated mass matrix and $\mathbf{D}$ the momentum remainder term. We denote the projection of the remainder momentum vector on the z-axis with the scalar d. $\mathbf{P}^*_K$ does now have a form similar to $\mathbf{P}_N$ and the joint velocity can be found, just like for body N:

$$\dot{\theta}_K = \frac{1}{J^*_{zz_K}} \left[(p_K - d_K) - \begin{pmatrix} 0 & \mathbf{e}_z^T \end{pmatrix}_K M^*_K \begin{pmatrix} \mathbf{v}_K \\ \boldsymbol{\omega}_{K-1} \end{pmatrix} \right] \tag{23}$$

p_K is the projection of the *articulated* angular momentum vector $\mathbf{p}^*_{a_k}$ on the local z-axis. In case of a fixed base, the linear speed $\mathbf{v}_1$ of point O_1 (see Fig. 1(a)) on the joint axis between the *fixed* base and the first body is zero. The angular velocity of the base is also zero. This allows for a very simple expression for the joint velocity at link 1, which can be calculated directly:

$$\dot{\theta}_1 = \frac{1}{J^*_{zz_1}}(p_1 - d_1) \tag{24}$$

The joint velocity can thereafter be used to compute the angular velocity ω_1 and the Cartesian speed $\mathbf{v}_2$, by means of the velocity transformation ${}^2\mathcal{T}_1^V$. These on their turn enable the calculation of the joint velocity $\dot{\theta}_2$ and so on. All joint velocities can be found by forward recursion. The obtained Cartesian and angular velocities are also used to compute the first set of Hamiltonian equations derived in the previous section. So, in a first, backward recursion, the articulated mass matrices, the momentum remainder vectors and the accumulated forces are calculated. In a subsequent, forward recursion, the joint velocities and time derivatives of the canonical momenta are computed. Acceleration based algorithms typically need a third recursion step for the forward kinematics. This gives an additional advantage to the canonical momenta based method, when implemented on a parallel computing architecture.

In the case of a fixed base, a thorough inspection of the algorithm revealed a maximum of 363 operations are needed for each body. Due to simplifications at the first and last bodies, this amount is reduced with at least 475 operations for the complete mechanism. This can be written: $363n - 475$, with n the number of bodies (degrees of freedom). This formula is applicable for $n \geq 3$. For comparison, a list of acceleration based algorithms and their amount of operations is shown in following tabel.

Algorithm	Additions	Multiplications	Total
Featherstone [4]	$275n - 18$	$336n - 220$	$611n - 238$
Vukobratović [10]	$231n - 294$	$249n - 272$	$480n - 566$
Rein [8]	$195n - 247$	$216n - 317$	$411n - 669$
Canonical momenta	$178n - 230$	$185n - 245$	$363n - 475$

6 Conclusions

In this paper, a recursive $O(n)$ algorithm has been introduced for the derivation of a set of Hamiltonian equations. The method is very promising compared to acceleration based algorithms thanks to: a reduced number of arithmetical operations needed to obtain the equations of motion, a potentially advantageous behavior during numerical integration and a reduced number of recursion steps.

References

[1] Dae-Sung Bae, Jon G. Kuhl and Edward J. Haug. A Recursive Formulation for Constrained Mechanical System Dynamics: Part III. Parallel Processor Implementation. *Mechanics of Structures and Machines*, 16(2):249–269, 1988.

[2] E. Bayo, A. Avello. Singularity-Free Augmented Lagrangian Algorithms for Constrained Multibody Dynamics. *Nonlinear Dynamics*, 5:209–231, 1994.

[3] K.E. Brenan, S.L. Campbell and L.R. Petzold. *The Numerical Solution of Initial Value Problems in Differential-Algebraic Equations*. Elsevier Science Publishing, 1989.

[4] Roy Featherstone. *Robot Dynamics Algorithms*. Kluwer Academic Publishers, 1987.

[5] Javier García de Jalón, Eduardo Bayo. *Kinematic and Dynamic Simulation of Multibody Systems: The Real-Time Challenge*. Springer-Verlag, New York, 1994.

[6] Herbert Goldstein. *Classical Mechanics*. Addison-Wesley Publishing Company, 1950.

[7] Thomas R. Kane and David A. Levinson. *Dynamics: Theory and Applications*. McGraw-Hill, 1985.

[8] U. Rein. Efficient Object Oriented Programming of Multibody Dynamics Formalisms. *Proceedings of the NATO-Advanced Study Institute on Computer Aided Analysis of Rigid and Flexible Mechanical Systems*, 2:59–69, 1993.

[9] Dan E. Rosenthal. An Order n Formulation for Robotic Systems. *The Journal of the Astronautical Sciences*, 38(4):511–529, October-December 1990.

[10] M.K. Vukobratović, V.F. Filaretov and A.I. Korzun. A Unified Approach to Mathematical Modelling of Robotic Manipulator Dynamics. *Robotica*, 12:411–420, 1994.

DEVELOPMENT OF PLATE AND SHELL ELEMENTS FOR FLEXIBLE MULTIBODY APPLICATIONS

AKI MIKKOLA
Lappeenranta University of Technology
Skinnarilankatu 34, 53851 Lappeenranta, Finland

AHMED SHABANA
University of Illinois at Chicago
842 West Taylor Street, Chicago, Illinois 60607, USA

1. Introduction

Development of geometrically nonlinear plate and shell elements for multibody analysis has been the subject of many investigations. Existing finite element solution procedures for large rotation and deformation analysis can be categorized as the incremental approach [1] or the large rotation vector approach [2, 3]. The incremental methods, that were developed for conventional non-isoparametric plate and shell elements, employ infinitesimal rotations to define the configuration of the finite element in the global inertial frame of reference. This approach leads to linearization of the rigid body kinematic equations, and as a consequence the description of rigid body displacements may not be exact [4]. In order to overcome this problem, several large rotation vector formulations were recently proposed. In these formulations, finite rotation parameters are used as nodal coordinates. Continuity conditions are imposed on the displacements and the rotation parameters at the element nodes. However, continuity of the finite rotations at the nodal points does not guarantee the continuity of the displacement gradients at these points. As a result, the centerline or the mid-surface of the element is not smooth. Therefore, the obtained solution eventually leads to errors in the calculations of the elastic forces and stresses at the nodal points. The large rotation vector formulations require interpolation of rotations which must be carefully handled, particularly in three-dimensional applications.

Due to above mentioned facts, existing finite element formulations for multibody problems are typically used in the framework of incremental solution procedures. As pointed out by Sharf [5], the incremental procedure is cumbersome to use in multibody analysis since forces acting on each flexible body are usually not all known. Moreover, linearization of finite rotations leads to incorrect integrals of motion and energy drift [6]. Therefore, there is a need to develop a new method for large deformation and rotation analysis of plates and shells that does not lead to a linearization of the dynamic equations and leads to the correct integral of motion.

The objective of this study is to present a new finite plate and shell elements for the multibody analysis based on the absolute nodal coordinate formulation. The absolute

nodal coordinate formulation was recently developed for large deformation and large rotation problems. In the absolute nodal coordinate formulation, only global displacement and slope coordinates are used as nodal variables, thereby avoiding difficulties that arise when rotations are interpolated in three-dimensional applications. By using slopes instead of rotations, no assumptions are made with regard to the magnitude of the rotations or the deformation within the element. Moreover, the use of slope coordinates ensures continuity of the rotations of the cross section as well as all the displacement gradients at the nodal points. The formulation can be used systematically to relax some of the assumptions used in the classical Kirchhoff and Mindlin plate models. Unlike other existing finite element formulations that lead to highly nonlinear inertial forces for three-dimensional elements, the absolute nodal coordinate formulation leads to a constant mass matrix, and as a result, the centrifugal and Coriolis inertia forces are identically equal to zero. This important property remains in effect even in the case of flexible bodies with slope discontinuities.

2. Geometric and Kinematic Descriptions of the Finite Element

In the absolute nodal coordinate formulation, the shape function matrix and the nodal coordinates can be used to define the element rigid body motion in the global coordinate system. Therefore, it is not necessary to use transformation between an element local coordinate system and the global coordinate system when the element configuration is defined. In the absolute nodal coordinate formulation, the global position vector $\mathbf{r}^{ij}$ of an arbitrary point P on an element j of the deformable body i can be written in the global coordinate system as follows:

$$\mathbf{r}^{ij} = \mathbf{S}^{ij}(x^{ij}, y^{ij}, z^{ij})\mathbf{e}^{ij} \qquad (1)$$

where $\mathbf{S}^{ij}$ is the element shape function matrix, $\mathbf{e}^{ij}$ is the vector of absolute nodal coordinates, x^{ij}, y^{ij} and z^{ij} are the spatial coordinates defined in the element coordinate system $\bar{\mathbf{x}}^{ij}$. The global definition of the shape function matrix can be achieved by using global displacements and slopes as nodal coordinates. By using the slopes as nodal coordinates instead of rotations, no assumptions are made with regard to the magnitude of the rotation or deformation within the finite element. The use of slopes also circumvents the difficulties that arise when a rotation or unit vector is interpolated in three-dimensional applications. In Eq. 1, the element is described as a continuous volume, making it possible to relax the assumption of rigid cross sections. Therefore, in large deformation problems the element cross section may deform and change its shape.

The plate element used in this investigation has four nodes each of which has 12 coordinates. The coordinates $\mathbf{e}^{ijk}$ of a node k on the element j of the deformable body i can be chosen as follows:

$$\mathbf{e}^{ijk} = \left[\mathbf{r}^{ijkT} \quad \left(\frac{\partial \mathbf{r}^{ijk}}{\partial x^{ij}}\right)^T \quad \left(\frac{\partial \mathbf{r}^{ijk}}{\partial y^{ij}}\right)^T \quad \left(\frac{\partial \mathbf{r}^{ijk}}{\partial z^{ij}}\right)^T \right]^T \qquad (2)$$

where the vector $\mathbf{r}^{ijk}$ defines the global position vector of node k and the three vectors $\frac{\partial \mathbf{r}^{ijk}}{\partial x^{ij}}$, $\frac{\partial \mathbf{r}^{ijk}}{\partial y^{ij}}$ and $\frac{\partial \mathbf{r}^{ijk}}{\partial z^{ij}}$ define the position vector gradients of node k. The shape function matrix $\mathbf{S}^{ij}$ can be derived by employing a polynomial expansion for the assumed

displacement field or by using other methods that are commonly used in the finite element literature [7]. As an example, the shape function matrix $\mathbf{S}^{ij}$ can be written as:

$$\mathbf{S}^{ij} = [S_1\mathbf{I} \;\; S_2\mathbf{I} \;\; S_3\mathbf{I} \;\; S_4\mathbf{I} \;\; S_5\mathbf{I} \;\; S_6\mathbf{I} \;\; S_7\mathbf{I} \;\; S_8\mathbf{I} \;\; S_9\mathbf{I} \;\; S_{10}\mathbf{I} \;\; S_{11}\mathbf{I} \;\; S_{12}\mathbf{I} \;\; S_{13}\mathbf{I} \;\; S_{14}\mathbf{I} \;\; S_{15}\mathbf{I} \;\; S_{16}\mathbf{I}] \tag{3}$$

where $\mathbf{I}$ is a 3 x 3 identity matrix and

$$S_1 = (2\xi+1)(\xi-1)^2(2\eta+1)(\eta-1)^2,\; S_2 = a\xi(\xi-1)^2(2\eta+1)(\eta-1)^2,\; S_3 = b\eta(\xi-1)^2(2\xi+1)(\eta-1)^2,$$

$$S_4 = t\zeta(\xi-1)(\eta-1),\; S_5 = -\xi^2(2\xi-3)(2\eta+1)(\eta-1)^2,\; S_6 = a\xi^2(\xi-1)(2\eta+1)(\eta-1)^2,$$

$$S_7 = -b\eta\xi^2(2\xi-3)(\eta-1)^2,\; S_8 = -t\xi\zeta(\eta-1)\;\; S_9 = \eta^2\xi^2(2\xi-3)(2\eta-3),$$

$$S_{10} = -a\eta^2\xi^2(\xi-1)(2\eta-3),\; S_{11} = -b\eta^2\xi^2(\eta-1)(2\xi-3),\; S_{12} = t\xi\zeta\eta,$$

$$S_{13} = -\eta^2(2\xi+1)(\xi-1)^2(2\eta-3),\; S_{14} = -a\xi\eta^2(\xi-1)^2(2\eta-3),\; S_{15} = b\eta^2(\xi-1)^2(2\xi+1)(\eta-1),$$

$$S_{16} = -t\eta\zeta(\xi-1)$$

where $\xi = x/a$, $\eta = y/b$ and $\zeta = z/t$. In the preceding equation, a, b, and t are the length, width and thickness of the plate element, respectively.

3. Slope Discontinuities

The position vector gradients can be evaluated using any sets of parameters. In order to be able to model slope discontinuities using simple linear connectivity conditions that lead to a constant mass matrix for the element that undergoes finite rotation and an arbitrary large deformations, a *body parameterization* instead of the local element parameters is used. In this representation, the vector $\mathbf{e}^{ijk}$ in Eq. 2 is expressed in terms of the body parameters X^i, Y^i and Z^i. The body coordinates are defined in a selected body coordinate system that represents a unique standard for all the finite elements of this body as shown in Figure 1. Without any loss of generality, the axes of this body coordinate system can be selected to be initially parallel to the axes of the global inertial frame of reference.

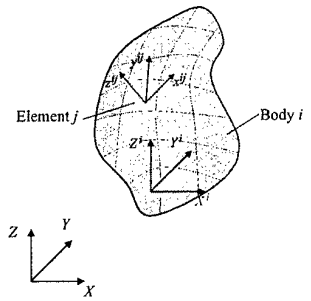


Figure 1. Element and body coordinate systems.

In order to deal with slope discontinuities between the finite elements, the transformation that relates the local element parameters $\bar{\mathbf{x}}^{ij}$ to the body parameters $\mathbf{X}^i$ need to be obtained. To this end, we note that:

$$\frac{\partial r_m^{ijk}}{\partial x_n^{ij}} = \frac{\partial r_m^{ijk}}{\partial X^i}\frac{\partial X^i}{\partial x_n^{ij}} + \frac{\partial r_m^{ijk}}{\partial Y^i}\frac{\partial Y^i}{\partial x_n^{ij}} + \frac{\partial r_m^{ijk}}{\partial Z^i}\frac{\partial Z^i}{\partial x_n^{ij}} \qquad m,n = 1,2,3 \tag{4}$$

where x_n^{ij} is the n th component of the vector $\bar{\mathbf{x}}^{ij}$ and r_m^{ijk} is the m th component of

vector $\mathbf{r}^{ijk}$. The preceding equation leads to nine scalar equations that define the transformation between the two sets of position vector gradients. Using these nine equations, the following transformation for the element coordinates can be obtained:

$$\mathbf{e}^{ijk} = \begin{bmatrix} \mathbf{r}^{ijkT} \\ \frac{\partial \mathbf{r}^{ijk}}{\partial x^{ij}} \\ \frac{\partial \mathbf{r}^{ijk}}{\partial y^{ij}} \\ \frac{\partial \mathbf{r}^{ijk}}{\partial z^{ij}} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & j_{1,1}^{ijk}\mathbf{I} & j_{2,1}^{ijk}\mathbf{I} & j_{3,1}^{ijk}\mathbf{I} \\ \mathbf{0} & j_{1,2}^{ijk}\mathbf{I} & j_{2,2}^{ijk}\mathbf{I} & j_{3,2}^{ijk}\mathbf{I} \\ \mathbf{0} & j_{1,3}^{ijk}\mathbf{I} & j_{2,3}^{ijk}\mathbf{I} & j_{3,3}^{ijk}\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{r}^{ijkT} \\ \frac{\partial \mathbf{r}^{ijk}}{\partial X^{i}} \\ \frac{\partial \mathbf{r}^{ijk}}{\partial Y^{i}} \\ \frac{\partial \mathbf{r}^{ijk}}{\partial Z^{i}} \end{bmatrix} = \mathbf{T}^{ijk}\mathbf{p}^{ijk} \qquad (5)$$

where $\mathbf{p}^{ijk}$ is the vector of coordinates of the nodal point k on element j of body i defined using the body parameterization, $\mathbf{I}$ is a 3 x 3 identity matrix, $\mathbf{T}^{ijk}$ is the transformation matrix that relates the two sets of the nodal coordinates at the nodal point k, and $j_{l,n}^{ijk}$ is the (l,n) th element of the Jacobian matrix that defines the relationship between the displacement gradients in the undeformed configuration. In the absolute nodal coordinate formulation, the vector $\mathbf{X}^{ijk} = \begin{bmatrix} X^{ijk} & Y^{ijk} & Z^{ijk} \end{bmatrix}^T$ that defines the global displacements of the nodal point k in the undeformed configuration can simply be written as

$$\mathbf{X}^{ijk} = \mathbf{S}^{ij}(x^{ijk}, y^{ijk}, z^{ijk})\mathbf{e}_0^{ij} \qquad (6)$$

where x^{ijk}, y^{ijk} and z^{ijk} are the local coordinates that define the position of the nodal point k in the element coordinate system, and $\mathbf{e}_0^{ij}$ is the vector of nodal coordinates in the initial configuration. The (l,n) th element of the Jacobian matrix can then be simply written as:

$$j_{l,n}^{ijk} = \left[\frac{\partial \mathbf{S}_l(x^{ijk}, y^{ijk}, z^{ijk})}{\partial x_n^{ij}} \right] \mathbf{e}_0^{ij} \qquad (7)$$

where $\mathbf{S}_l$ is the l th row of the shape function matrix. For a finite element that consists of n_n nodes, the element transformation matrix can be written as follows:

$$\mathbf{T}^{ij} = \begin{bmatrix} \mathbf{T}^{ij1} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbf{T}^{ij(n_n)} \end{bmatrix} \qquad (8)$$

Using Eqs. 1 and 5, the element configuration in the global coordinate system can be expressed as follows:

$$\mathbf{r}^{ij} = \mathbf{S}^{ij}\mathbf{T}^{ij}\mathbf{p}^{ij} \qquad (9)$$

As previously shown [8], Eq. 9 can be used to describe an arbitrary rigid body motion if the transformation matrix $\mathbf{T}^{ij}$ remains constant while changes are made in the vector $\mathbf{p}^{ij}$.

4. Dynamic Equations

The inertia matrix of the element j can be calculated using the following expression of the kinetic energy:

$$T^{ij} = \frac{1}{2}\int_{V^{ij}} \rho^{ij}\, \dot{\mathbf{r}}^{ijT}\dot{\mathbf{r}}^{ij}\, dV^{ij} \qquad (10)$$

where ρ^{ij} is the mass density, V^{ij} is the volume and $\dot{\mathbf{r}}^{ij}$ is the absolute velocity vector of the element j. Using the transformation matrix that accounts for the slope discontinuities, the absolute velocity vector can be written as follows:

$$\dot{\mathbf{r}}^{ij} = \mathbf{S}^{ij}\mathbf{T}^{ij}\dot{\mathbf{p}}^{ij} \tag{11}$$

By substituting Eq. 11 into the expression of the kinetic energy, one obtains:

$$T^{ij} = \frac{1}{2}\dot{\mathbf{p}}^{ijT}\mathbf{M}^{ij}\,\dot{\mathbf{p}}^{ij} \tag{12}$$

where $\mathbf{M}^{ij}$ is the element mass matrix defined as:

$$\mathbf{M}^{ij} = \mathbf{T}^{ijT}\left[\int_{V^{ij}} \rho^{ij}\,\mathbf{S}^{ijT}\mathbf{S}^{ij}\,dV^{ij}\right]\mathbf{T}^{ij} \tag{13}$$

The mass matrix $\mathbf{M}^{ij}$ depends on the mass density, transformation matrix $\mathbf{T}^{ij}$ and dimensions of the element. Since $\mathbf{J}_0^{ij}$ is a constant matrix whose elements are defined using Eq. 7, the mass matrix remains constant despite the discontinuities of the slopes and the initial curvature of the element.

In the absolute nodal coordinate formulation, two different methods can be used when the elastic forces within the finite element are defined. In the first approach for definition of the elastic forces, a local coordinate system is employed to define the element deformations. The use of the local element coordinate system leads, as demonstrated in previous publications, to a more complex expression for the elastic forces [10]. This approach is not employed in this investigation. A straightforward approach for evaluating the elastic forces is to use a continuum mechanics approach. In this case there is no need for defining the element deformation in a local element coordinate system. This approach leads to the general expression of the elastic forces since the nonlinear strain-displacement relationship must be used in order to avoid spurious strains. Using a continuum mechanics approach, the global displacement gradients can be obtained directly as:

$$\mathbf{D}^{ij} = \mathbf{J}^{ij}\left[\mathbf{J}_0^{ij}\right]^{-1} = \frac{\partial\left(\mathbf{S}^{ij}\mathbf{T}^{ij}\,\mathbf{p}^{ij}\right)}{\partial\bar{\mathbf{x}}^{ij}}\left[\frac{\partial\left(\mathbf{S}^{ij}\mathbf{T}^{ij}\,\mathbf{p}_0^{ij}\right)}{\partial\bar{\mathbf{x}}^{ij}}\right]^{-1} \tag{14}$$

where $\mathbf{p}_0^{ij}$ is the vector of nodal coordinates that defines the element initial configuration in the body coordinate system. The strain tensor can be obtained using the matrix $\mathbf{D}^{ij}$ and the Cauchy-Green formula as follows:

$$\epsilon_m^{ij} = \frac{1}{2}\left(\mathbf{D}^{ijT}\mathbf{D}^{ij} - \mathbf{I}\right) \tag{15}$$

where $\mathbf{I}$ is a 3x3 identity matrix. The elastic forces of the finite element j can be derived by using the principle of virtual work as follows:

$$\delta W^{ij} = -\int_{V^{ij}} \epsilon^{ijT}\mathbf{E}^{ij}\delta\epsilon^{ij}\,dV^{ij} \tag{16}$$

where $\mathbf{E}^{ij}$ is the matrix of elastic coefficients, and ϵ^{ij} is the vector form of the strain tensor ϵ_m^{ij}.

5. Summary and Conclusions

A new plate and shell elements developed using the absolute nodal coordinate

formulation are presented in this investigation. In the proposed element formulation, only global displacements and slope coordinates are used as nodal variables. The proposed formulation circumvents difficulties that arise when rotations are interpolated. The absolute nodal coordinate formulation uses a displacement field that defines the location of the arbitrary points on the plate or shell element in the global system, not in an element coordinate system. Since a displacement field is linear in the nodal coordinates, the absolute nodal coordinate formulation leads to a constant mass matrix as demonstrated in this study.

The general plate and shell element developed in this investigation can describe rigid body motion, finite rotations and an arbitrary large deformation. Continuity of all displacement gradients at the element mid-surfaces are ensured, thereby, ensuring the smoothness of the mid-surface of the structure model. A continuum mechanics approach with nonlinear strain-displacement relationships is used to obtain the plate elastic forces that account for all geometric nonlinearities and shear deformation. The proposed formulation is fundamentally different from the three-dimensional degenerated element since the proposed element contains information about all the rotational degrees of freedom at the nodes. This property allows using this new formulation in the framework of a non-incremental solution procedure. It is shown in this paper that the property of the constant mass matrix remains in effect when the absolute nodal coordinate formulation is used to model flexible bodies with slope discontinuities.

6. References

1. Rankin, C.C., and Brogan, F.A. (1986) An Element Independent Corotational Procedure for the Treatment of Large Rotations, *ASME Journal of Pressure Vessel Technology*, **108**, pp. 165-174.
2. Simo, J.C., and Vu-Quoc, L. (1986) On the Dynamics of Flexible Beams Under Large Overall Motion – the Plane Case: Part I, *ASME Journal of Applied Mechanics*, **53**, pp. 849-854.
3. Simo, J.C., and Vu-Quoc, L. (1986) A Three-Dimensional Finite-Strain Rod Model. Part II: Computational Aspects, *Computer Method in Applied Mechanics and Engineering*, **58**, pp. 79-116.
4. Shabana, A.A. (1996) Finite Element Incremental Approach and Exact Rigid Body Inertia, *ASME Journal of Mechanical Design,* **118**, pp. 171-178.
5. Sharf, I. (1999) Nonlinear strain measures, shape functions and beam elements for dynamics of flexible beams," *Multibody System Dynamics*, **3**, pp. 189-205.
6. Shabana, A.A., and Mikkola, A.M., On the use of the degenerated plate and the absolute nodal coordinate formulations in multibody system applications, *Journal of Sound and Vibration* (in printing)
7. Bathe, K.J. (1996) *Finite Element Procedures,* Prentice-Hall.
8. Shabana A.A. and Mikkola, A. M. (2002) Modeling of Slope Discontinuities in Flexible Body Dynamics Using the Finite Element Method, *Proceedings of the 2002 ASME International Design Engineering Technical Conferences*, Montreal, Canada, September 29 - October 2.
9. Shabana A.A. (1998) *Dynamics of Multibody Systems,* 2nd edition, Cambridge University Press.

PARALLEL COMPUTING IN THE CONTEXT OF MULTIBODY SYSTEM DYNAMICS

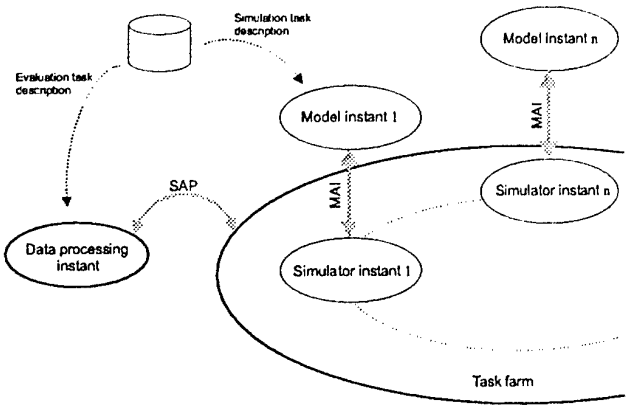
ANDREAS MULLER
Institute of Mechatronics
at the Chemnitz University of Technology
Reichenhainer Str. 88, 09126 Chemnitz, Germany
Andreas.Mueller@ifm.tu-chemnitz.de

1. Distributed simulation of MBS dynamics

The need for high performance simulation of the dynamics of large MBSs is a widely recognized issue stimulated by demands from a variety of different application areas such as interactive real time virtual reality simulation, model based control and of course the design and development process. In particular the development of complex mechatronics systems calls for highly flexible simulation tools which are reconfigurable and model independent. Several interactive tools for the simulation of MBS dynamics exist (Adams, alaska, NewEul, Mobile), which are commonly intended to support the design process of one particular model but not for case studies, model fitting or even MBS optimizations. Approaches to the optimization of complex systems have always been tailor made implementation specific to the problem at hand. A general treatment was not attempted yet.

The use of parallel computing facilities (PCF) is well established for the numerical simulation of continuum mechanical, fluid mechanical as well as electromagnetic field problems. This is because the large number of degrees of freedom of the mathematical models can be immediately distributed on a parallel computing grid. However, though PCF have not been seriously employed in the context of MBS simulations, PCF are also potentially advantageous in many respects for the MBS dynamics simulation and optimization. The classical single-model/single-processor simulation systems may be extended to evaluate the instantaneous kinematics, kinetics and dynamics exploiting the MBS topology for a distributed evaluation of the motion equations employing very time efficient parallel $O(n)$ algorithms [2]. But the necessary computing resources are not justifiable since it could

only speed up the dynamics simulation of one MBS model. On the other hand the entire dynamics of one MBS model can be simulated per processing node. In this way autonomous running MBS models constitute **a task farm**. A combination of both approaches, i.e. model instants on this task farm use parallel $O(n)$ algorithms, is usually not possible with the current state of the art technology.



Such a task farm plus a superordinated controller/processing instant constitute a **MBS simulation grid**. The aim of the simulation grid is to provide MBS simulation results and incorporate these in data processing tasks. Since each node on the task farm can be considered as a stand-alone simulation tool this methodology shall feature the complete functionality of established simulation packages. Consequently a simulation may be every combination of possible task that a simulation package could perform, e.g. kinematic/dynamic simulation, linear analysis or equilibrium determination. In this way it is possible to perform full simulations of several instances of a parameterized MBS model in parallel.

One single controller/processing instant governs the task farm and serves the model instants on that task farm with necessary parameters. Incoming simulation results are processed by the controller/processing instant, they could simply be stored for later use or model parameters could be optimized to achieve a desired behavior.

A simulation grid for MBS was developed at the Edinburgh Parallel Computing Center (EPCC), Edinburgh, Scotland, UK in cooperation with the University of the Federal Armed Forces, Hamburg, Germany. The MBS modelling is supported by the interactive modelling and simulation system alaska. This system provides MBS models in a suitable form for the task farm.

2. Components of a distributed simulation environment

2.1. CONTROLLER AND PROCESSING INSTANT (CPI)

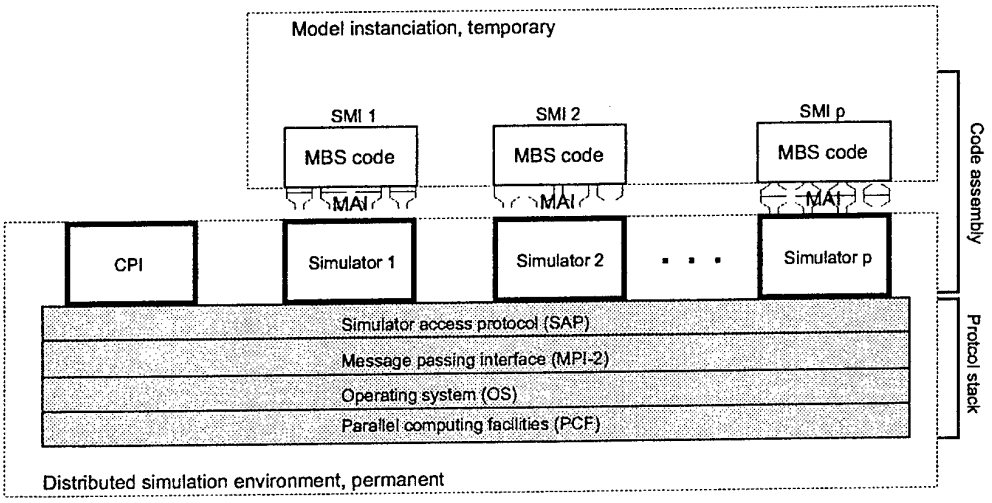
The controller/processing instant (CPI) controls the overall simulation grid. It is the only instant of the simulation grid accessible from the external

environment. The CPI is the actual data processing unit that provides the task farm with model and simulation parameters. MBS simulation results are obtained by the simulator instants, the data processing is accomplished by the single CPI.

Typical examples for data processing strategies are the parameter variation (collection of simulation results for different parameters) and the optimization of MBS kinematics or dynamics with respect to specified model parameters. From the CPI point of view the task farm entries are simply parameterized input-output relations. Thus the task farm entries may be any instants, not necessary MBS simulators, compliant to the communication framework described below.

2.2. SIMULATOR AND MODEL INSTANT (SMI)

Each simulator/model instant (SMI) placed on one node of the farm is a composite of the specific MBS model and a simulation engine. Here MBS model means C-code which is generated by an interactive simulation tool. In this way the code fulfills interface specifications in order to ensure model independence. The accompanying simulator can be considered as the simulation kernel of a standard simulation packages so that it is able to carry out the same simulation tasks as a user might do interactively. The problem specific simulation tasks are accomplished by the SMI in batch simulation mode and described by a command file which is common to all SMIs on the farm. As such the SMI has the full simulation functionality of classical stand-alone simulation tools except their interactive modelling capabilities. The SMIs appear to the PCI as black boxes and the CPI strictly has no information about the particular simulations carried out the PCIs.



2.3. SIMULATOR ACCESS PROTOCOL (SAP)

One of the main challenges of the developed system is the coordination of SMIs by the PCI. For the sake of flexibility and generality the PCI and SMIs, each being an individual process on a PCF node, communicate via a simulator access protocol (SAP). This SAP is based on the MPI-2 framework for distributed computing systems [8]. The SAP approach ensures a maximum flexibility of the system because the actual PCI as well as the SMIs can be freely substituted as long as they are compliant with the SAP specification. Consequently the data processing of developed simulation grid is not limited in type and complexity, i.e. parameter variation and optimization of MBS are two applications only. It may further cater for the distributed simulation of cooperating systems.

2.4. MODEL ACCESS INTERFACE (MAI)

The SMIs consist of two parts the MBS model and the simulation engine. While the PCI and SMIs are coupled via the SAP (a software protocol) the MBS (the model C-code) must be linked to the simulation engines (C-library). The general conditions for this interconnection are defined by a model access interface (MAI). This is nothing but a predefined set of C-functions with defined calling conventions. Any MBS model fulfilling this MAI convention can be linked to the simulator, which are of course MAI compliant.

3. MBS modeling and code generation

Crucial for an easy and straight forward implementation of MBS models on the task farm is the automatic generation of C-code fulfilling the MAI specification. The automatic generation of the model code has several advantages in terms of transparency, modularity and safety. One condition on the model description to facilitate this is a modular, or consequently object oriented, modeling [3,6]. Another condition is that the modelling and simulation tool that engineers use for interactive simulations is able to 'dump' its internal program flow for that MBS at hand in form of portable C-code which is also compliant with the MAI specification. This claim was achieved during the development of the simulation tool box alaska (www.tu-chemnitz.de/ifm).

4. Application: Nanometer coordinate measuring machine

The developed system was employed for the optimization of a fairly complex high precision coordinate measurement device [7] as part of the development process. A nanometer coordinate measuring machine (NCMM) combines the precision of nanometer measuring devices and the large workspace of conventional measuring machines. This is achieved by a novel cascadable setup. The machine is equipped with an atomic force microscope (AFM) as topography sensor. Obviously, the large scan volume contradicts the targeted high precision taking into account the used mechanical components. That is, the NCMM construction demands the use of high quality components to ensure high mechanical precision and the drive control must be able to rapidly reach a target position with very high accuracy. It turned out that the control of the NCMM is crucial and cannot be optimized by trial-and-error. Therefore the positioning system of an existing NCMM prototype was modelled as electromechanical rigid multibody system model with the alaska simulation package.

The controller parameter of the (existing) protype were optimized using a genetic algorithm [4,5]. The typical population consisted of 127 SMIs, i.e. 127+1 processors were in use. The main objective was to minimize the overshooting effect during positioning of the AFM tip. The optimization goal was achieved after less then 40 cycles and the optimal controller parameter constellation now yields 70% less overshooting during the tip approach. Also the scan motion precision could be improved.

References

1. DeJong, K.A. (2000) Adaptive system design: a genetic approach, *IEEE Trans. Syst., Man. and Cyber.*, **Vol. SMC-10 no. 9**
2. S. Duan, K.S. Anderson (2000) Parallel implementation of a low order algorithm for dynamics of multibody systems on a distributed memory computing system, *Engineering with Computers*, **Vol. 16**
3. A. Keil, *et. al* (1999) On the description of multibody system models, *Proc. EUROMECH Colloquium 404, Lisbon*
4. A. Müller, H. Rothe (2001) Optimization of controller parameter of a coordinate measuring machine, *Proc. 6. US national congress on computational mechanics (USNCCM) 2001, Dearborn, Detroit, USA*
5. A. Müller, A. Keil, H. Rothe, R. Petersen (2002) The dynamics of a coordinate measuring machine, *Proc. 1. International symposium on mechatronics (ISOM) 2002, Chemnitz, Germany*
6. U. Neerparsch (1996) Zur Standardisierung der Modellbeschreibung von Mehrkörperformalismen, *VDI-Berichte*, **no. 235**
7. H. Rothe, D. Hüser, and R. Petersen (2000) Simulation of the dynamic behaviour of a nanometer coordinate measuring machine, *K. Hasche, W. Mirandé, and G. Wilkening, (ed.), Proceedings of the 4th Seminar on Quantitative Microscopy: Dimensional measurements in the micro- and nanometer range*, **Vol. PTB-F-39**
8. Message Passing Interface Forum (1994), MPI: A Message-Passing Interface Standard, *International Journal of Supercomputer Applications*, **Vol. 8 no. 3/4**
   For MPI related informations refer to http://www.mpi-forum.org

COLLISION DETECTION AND ADMINISTRATION FOR MANY COLLIDING BODIES

B. MUTH, and P. EBERHARD
Institute B of Mechanics, University of Stuttgart, Pfaffenwaldring 9, D-70550 Stuttgart, Germany
bm@mechb.uni-stuttgart.de,
Institute of Applied Mechanics, University of Erlangen-Nuremberg, Egerlandstr. 5, D-91058 Erlangen, Germany
eberhard@ltm.uni-erlangen.de

1. Introduction

The paper deals with the calculation and administration of the motion and the contacts of systems that are comprising many colliding bodies. Special attention is paid to the comparison of the efficiency of the employed algorithms with respect to calculation time. In order to model the behavior of many particles very efficiently, methods from Molecular Dynamics (MD) are used. To reduce the high calculation time that is usually spend on the collision detection, sophisticated sorting algorithms for neighborship search are required. These algorithms are exerted before determining the contact forces applied to the particles. This holds especially for large systems with many repeatedly colliding particles. In the paper three of such method are discussed and compared for both the 2D and the 3D case. In order to determine the dynamical behavior of systems consisting of several or rather many particles, some fully developed approaches exist. Systems consisting of bodies with negligible deformations can be described e.g. by means of the multibody system method (MBS), (Schiehlen, 1986). Mass point systems may be regarded as a special case of MBS.
For studies of flexible bodies, usually the Finite-Element-Method (FEM) or the Boundary-Element-Method (BEM) are used, compare (Eberhard, 2000). Each of these methods has its own advantages and disadvantages. While the MBS is in general characterized by comparatively short computation times due to a small number of degrees of freedom, traditionally deformations cannot be handled. On the other hand, systems investigated

by the FEM possess a large number of degrees of freedom with rather complex equations of motion, but deformations are taken into account.
An expansion of the MBS method for elastic bodies is e.g. shown in (Melzer, 1994). Hybrid MBS / FEM contact calculations are presented in (Eberhard, 2000), where colliding bodies are examined by the FEM approach in order to incorporate deformations while all the other bodies of the system are regarded as rigid. This approach combining FEM and MBS makes use of the advantages of both of the methods. However, all of these mentioned approaches have one significant drawback for particle systems in common, in fact that the number of contacting particles is quite limited.
Very efficient methods for granular matter exist, that allow the determination of motions and contacts of many thousands of particles. For the efficient determination especially of systems consisting of a very large number of small elements, methods from MD are frequently used. The formulation of the contact forces between the different bodies is based on simple models in order to keep the calculation times within a feasible range. Here usually very small penetrations between the particles are accepted, see e.g. (Luding, 1998). Forces applied to a particle in MD are for instance gravitational forces and contact forces resulting from the boundaries of the system and from other particles within the system. The normal contact forces acting in opposite direction to the occurring penetrations are modeled as elastic restoring forces. The force is proportional to the penetration of the particles (Luding, 1998). This corresponds to a penalty force or homogenization approach.
By means of MD basically arbitrary systems such as gas, fluids, molecules or charge carriers can be investigated (Luding, 2000). Apart from the above mentioned rejecting contact forces also attractive forces, so called long range correlating forces, may occur. Further, the potential energy of each particle generally also depends on all the other bodies of the system. For example for atomic particles this influence is called 'van der Waals' or 'London' dispersion, see e.g. (Rapaport, 1995).
In a system with attractive forces each particle influences all the other ones within the system. This means, that for a system consisting of n particles, the required calculation operations for the contact force computation will be of order $O(n^2)$. Especially for systems consisting of many particles this fact will cause very long calculation times and only very small time intervals can be simulated.
Usually the body distances that are decisive for the major amount of the occurring repulsive forces between two particles are calculated in a double loop over all particles. Therefore, the time to examine all bodies with respect to separation or contact is proportional to n^2, see (Allen and Tildesley, 1987), leading still to the mentioned $O(n^2)$ behavior. For the investigation

of solids the attractions between the particles can be neglected so that just the directly contacting and neighboring bodies have an influence on each other. In 1967 Verlet suggested a technique which utilized this fact in order to improve the calculation speed (Allen and Tildesley, 1987). The idea is to generate a list of the neighbors of every body in order to perform collision detection not for each existing pair of particles of the system, but only for neighboring bodies. These lists need to be updated only from time to time and not for every step of the dynamic calculation. Since then many different approaches have been developed, in order to reduce the calculation time by efficiently extracting the surrounding particles of the bodies.

2. Neighbor Search Methods

In our paper three methods are presented that can be used in order to find neighboring bodies of a particle efficiently. The first two of these methods, the Verlet neighbor list and the linked cells method, identify the neighboring particles of a body by regarding special regions of the system and considering all particles within the same region as neighbors. These methods can have high advantages, as for some systems it is possible to reduce the calculation operations down to an order of $O(n)$, compare (Muth, 2001). For both methods the neighboring zones have to be at least somewhat larger than the particles themselves, compare (Allen and Tildesley, 1987) or (Muth, 2001). Out of that, two problems can arise. Firstly, if the particles within the system are polydisperse, that means their sizes differ, then the size of the neighboring zones has to conform to the largest particle existing in the system. Hence, for highly polydisperse mixtures, the smaller particles may increase the average number n_c, which might in the worst case even be close to n, see (Schinner, 1998).
As the neighborship zones around the particles are larger than the particles, the neighbor lists do not have to be updated in each time step. The size of the zones and the necessary update frequency are interdependent and not quite easy to guess. Therefore, another problem of both methods is the ascertainment of optimal values for both, the update frequency for the lists and the size of the zones. If the zones are too small or the update time steps too large, the behavior of the system cannot be calculated correctly. But, if the zones are too large or the time steps for rebuilding the lists are too small, the calculations will be inefficient. The choice of these values is therefore very important, and it may take a lot of time getting experience with the investigated system.
The third method presented more detailed, the linked linear list method, is based on a totally different approach. It is also a very efficient method, used to keep track of neighbors for large systems. In a first step, bounding

boxes are laid around each particle, Fig. 1, that are seized in such a way, that each particle fits exactly in its box. The edges of each bounding box are aligned parallel to the system axes.

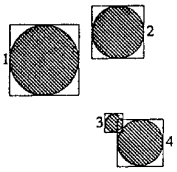


Figure 1. Bounding box around each particle.

In a next step the bounding boxes are projected separately onto the system axes. Such a projection onto the x-axis for the situation in Fig. 1 is shown in Fig. 2. In the following, only the order of the beginnings 'b' and endings 'e' of the projections of the bounding boxes along the axes is of interest. For this reason the sequences are stored in lists.

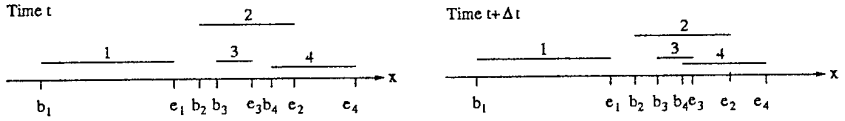


Figure 2. Particles projected on the x-axis for two different times, (Schinner, 1998).

For a 3D system, three different projections are necessary and, therefore, three lists will be compiled, each with the length of two times the numbers of particles in the system. If there is the beginning, ending, or both, of another particle in between the beginning and ending of a particular body, then there will be an overlap of the projections of the bounding boxes of both particles along this axis. A collision of two bounding boxes exists for an overlap of these projections along each axis.
Checking whether there is some part of a projection in between the beginning and ending of another projection for each particle along each axis still takes a lot of time. But, although these lists have to be updated for each time step, the necessary calculation times can be reduced to a number proportional to the total number of particles in the system, as there only has to be done an update instead of a complete recomputation of the old list for each new time step, that corresponds to sorting an already nearly sorted list. This update can simply be done by going through the lists sequentially and checking for any new changes in the order. The occurring changes usually only are permutations, compare e.g. Fig. 2, where e_3 and b_4 have been changed. If the order of the beginnings and endings does not have to be changed, the collision status of the particles also will remain unchanged. While seeking for new colliding bounding boxes, by looking

for permutations in the lists, four different cases have to be discerned, compare (Schinner, 1998).

- Two beginnings are changed, which means the bounding boxes have been overlapping and will continue to overlap,
- two endings are changed, which also means the bounding boxes have been overlapping and will continue to overlap,
- a beginning and a proximate ending are changed, which means a so far occurring overlap will now have to be removed or
- an ending and a following beginning of another particle are exchanged, which means there will now be a new overlap between their two bounding boxes.

For the first two cases nothing has to be done in the lists at all, as the collision status between any particle will not change. If a collision along an axis has to be removed, or if there is a new collision between two particles along an axis, the collision information along the other axes is essential, in order to know, whether there is a new collision along all axes and, therefore, between the bounding boxes or, on the other hand, there will now be no overlap any more between two so far colliding bounding boxes. For this reason for the 2D case a second and a third column are added to the lists, that store the information of the positions of beginnings and endings along the y-axis, compare Fig. 3. In each row, the positions are stored, of beginnings (column two) and endings (column three) of the particle of the first column. For a 3D system also a fourth and a fifth column have to be added with the position information of beginnings and endings along the z-axis.

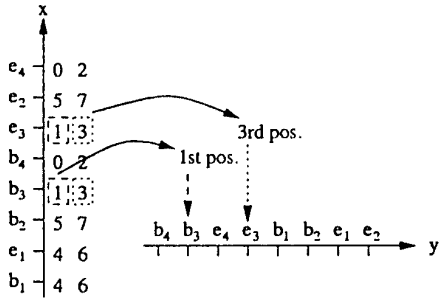


Figure 3. Lists containing also the position information along the other axes.

For example going through the list along axis x, see Fig. 3, leads to the potential collision between particles (2/3), (2/4), and (3/4). As the location of particle 3 along the y-axis is from position one to three, whereas the end of particle 4 has the position two, there is also an overlap of bounding boxes 3 and 4 along the y-axis and, therefore, real collision of the bounding boxes

of particles 3 and 4. Therefore, particle 3 and 4 are now considered to be neighbors.
For each method the neighboring particles are stored in lists. Thus, after the pre-sorting has been finished, the real collision detection needs to be done only for these surely neighboring and potentially colliding bodies. Hence, the necessary calculation operations for collision detection can be reduced down to an order proportional to the number of particles in the system, i.e. $O(n)$.
In the paper results for different examples are shown comparing the three described methods. It turns out that the Verlet neighborhood lists are always quite time consuming while there is no clear 'winner' from the other two methods. Depending on the density and polydispersity of the investigated system either method has advantages and disadvantages.

Acknowledgements

The authors want to acknowledge the help of Prof. S. Luding and Dipl.–Phys. M.-K. Müller during various discussions. They also provided the data for the linked cell method which are used in the comparison. We further thank the German Research Council (DFG) for supporting this project in the Collaborative Research Center (SFB) 404, project B7.

References

M.P. Allen and D.J. Tildesley. *Computer Simulation of Liquids.* Oxford: University Press, 1987.

P. Eberhard. *Kontaktuntersuchungen durch hybride Mehrkörpersystem / Finite Elemente Simulationen* (in German). Aachen: Shaker, 2000.

S. Luding. Collisions and Contacts Between Two Particles. *Physics of Dry Granular Media*, Nato ASI Series E350, pp. 285–304, 1998.

S. Luding. *Die Physik kohäsionsloser granularer Medien* (in German). Berlin: Logos, 1998.

F. Melzer. *Symbolisch-numerische Modellierung elastischer Mehrkörpersysteme mit Anwendung auf rechnerische Lebensdauervorhersagen* (in German). Düsseldorf: VDI Verlag, Dissertation, Fortschritt-Berichte Reihe 20, Nr.139, 1994.

B. Muth. Simulation von Kontaktvorgängen einfacher Körper mit Methoden der Molekulardynamik (in German), *Master Thesis*, DIPL–87, University of Stuttgart, Institute B of Mechanics, 2001.

D.C. Rapaport. *The Art of Molecular Dynamics Simulation.* Cambridge: University Press, 1995.

W. Schiehlen. *Technische Dynamik* (in German). Stuttgart: B.G. Teubner, 1986.

A. Schinner. Fast Algorithms for the Simulation of Polygonal Particles. *Granular Matter 2*, pp. 35–43, 1998.

ON THE ISSUE OF ITERATIVE LINEAR ALGORITHMS FOR THE MULTI-THREADED SIMULATION OF MECHANICAL SYSTEMS REPRESENTED IN CARTESIAN COORDINATES

DAN NEGRUT
MSC.Software
2300 Traverwood Drive, Ann Arbor, MI 48105, USA

1. Introduction

The generalized coordinates used in this paper to represent the state of a mechanical system are Cartesian coordinates for position, and Euler parameters for orientation of body centroidal reference frames; i.e., for body i, $\mathbf{r}_i = [x_i, y_i, z_i]^T$, and $\mathbf{p}_i = [e_{i0}, e_{i1}, e_{i2}, e_{i3}]^T$, respectively. The Euler parameters must satisfy the parameterization constraints, $\mathbf{p}_i^T \mathbf{p}_i = 1$, $1 \leq i \leq n_b$, [2], where n_b is the number of rigid bodies in the model. The vector of generalized coordinates is defined as $\mathbf{q} = \left[\mathbf{r}_1^T, \ldots, \mathbf{r}_{n_b}^T, \ \mathbf{p}_1^T, \ldots, \mathbf{p}_{n_b}^T\right]^T \in \Re^n$, $n = 7n_b$. To simplify the presentation it is assumed that the joints in the model only introduce holonomic constraints. The kinematic constraints at the position, velocity, and acceleration levels assume the expression

$$\mathbf{\Phi}(\mathbf{q}, t) = \left[\ \Phi_1(\mathbf{q}, t) \quad \ldots \quad \Phi_m(\mathbf{q}, t) \ \right]^T = \mathbf{0} \tag{1}$$

$$\mathbf{\Phi}_\mathbf{q}(\mathbf{q}, t)\, \dot{\mathbf{q}} + \mathbf{\Phi}_t(\mathbf{q}, t) = \mathbf{0} \tag{2}$$

$$\mathbf{\Phi}_\mathbf{q}(\mathbf{q}, t)\, \ddot{\mathbf{q}} + \left(\mathbf{\Phi}_\mathbf{q}(\mathbf{q}, t)\dot{\mathbf{q}}\right)_\mathbf{q} \dot{\mathbf{q}} + 2\mathbf{\Phi}_{\mathbf{q}t}(\mathbf{q}, t)\dot{\mathbf{q}} + \mathbf{\Phi}_{tt}(\mathbf{q}, t) = \mathbf{0} \tag{3}$$

If m_k represents the number of constraints induced by the joint k, then $\mathbf{\Phi}(\mathbf{q}, t) \in \Re^m$, where $m = \sum_{k=1}^{N_J} m_k$, with N_J denoting the number of joints present in the model. The subscript denotes partial differentiation, $\mathbf{\Phi}_\mathbf{q} = [\partial \Phi_i / \partial q_j]$, $i = 1, \ldots, m$, $j = 1, \ldots, n$. It is assumed that the m constraint equations are independent; i.e., $\mathbf{\Phi}_\mathbf{q}$ has full row rank.

In what follows, for body i, $\delta \mathbf{r}_i$ and $\delta \bar{\pi}_i$ represent a virtual translation and rotation, respectively, m_i is the mass of the body, $\mathbf{f}_i$ is the vector of applied forces, $\bar{\omega}_i$ is the angular velocity represented in the centroidal body-fixed reference frame, $\bar{\mathbf{J}}_i$ is the inertia tensor, and $\bar{\mathbf{n}}_i$ is the applied

torque expressed in the local reference frame. The notation convention that a vector quantity with an over-bar is represented in a local reference frame is observed in what follows. The Lagrange multiplier form of the equations of motion assumes the form

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{J}} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{r}} \\ \dot{\bar{\omega}} \end{bmatrix} + \begin{bmatrix} \eta^T(\mathbf{\Phi},\mathbf{q}) \\ \rho^T(\mathbf{\Phi},\mathbf{q}) \end{bmatrix} \lambda = \begin{bmatrix} \mathbf{f} \\ \hat{\mathbf{n}} \end{bmatrix} \tag{4}$$

where $\mathbf{M} = diag(\mathbf{M}_1, \ldots, \mathbf{M}_{n_b})$, $\bar{\mathbf{J}} = diag(\bar{\mathbf{J}}_1, \ldots, \bar{\mathbf{J}}_{n_b})$, $\ddot{\mathbf{r}} = [\ddot{\mathbf{r}}_1^T, \ldots, \ddot{\mathbf{r}}_{n_b}^T]^T$, $\dot{\bar{\omega}} = [\dot{\bar{\omega}}_1^T, \ldots, \dot{\bar{\omega}}_{n_b}^T]^T$, $\eta(\mathbf{\Phi},\mathbf{q}) = [\eta_1(\mathbf{\Phi},\mathbf{q}), \ldots, \eta_{n_b}(\mathbf{\Phi},\mathbf{q})]$, $\rho(\mathbf{\Phi},\mathbf{q}) = [\rho_1(\mathbf{\Phi},\mathbf{q}), \ldots, \rho_{n_b}(\mathbf{\Phi},\mathbf{q})]$, $\mathbf{f} = [\mathbf{f}_1^T, \ldots, \mathbf{f}_{n_b}^T]^T$, and $\hat{\mathbf{n}} = [\hat{\mathbf{n}}_1^T, \ldots, \hat{\mathbf{n}}_{n_b}^T]^T$, with $\hat{\mathbf{n}}_i = \bar{\mathbf{n}}_i - \tilde{\bar{\omega}}_i \bar{\mathbf{J}}_i \bar{\omega}_i$. Note that $\eta_i(\mathbf{\Phi},\mathbf{q})$ and $\rho_i(\mathbf{\Phi},\mathbf{q})$ are the linearization operators that in the expression of the first order variation of the position constraint equations of Eq.1 multiply the virtual translation $\delta\mathbf{r}_i$ and rotation $\delta\bar{\pi}_i$; i.e., $\delta\mathbf{\Phi} = \sum_{i=1}^{n_b} \eta_i(\mathbf{\Phi},\mathbf{q})\, \delta\mathbf{r}_i + \sum_{i=1}^{n_b} \rho_i(\mathbf{\Phi},\mathbf{q})\, \delta\bar{\pi}_i$, [4].

The numerical solution of the index 3 DAE of Eq.1 and 4 is found using an explicit integration formula that integrates a set of state space ordinary differential equations (SSODE). The DAE to SSODE reduction is based on a partitioning of the generalized positions $\mathbf{q}$ in dependent coordinates $\mathbf{u} \in \Re^m$, and independent coordinates $\mathbf{v} \in \Re^{ndof}$, $ndof = n - m$, [5]. The coordinate partitioning–based approach requires at each integration step the acceleration $\ddot{\mathbf{q}}$. Note that if λ is available, $\ddot{\mathbf{r}}$ and $\dot{\bar{\omega}}$ are expeditiously computed based on Eq.4, and with $\ddot{\mathbf{p}}_i = 0.5\, \mathbf{G}_i^T \dot{\bar{\omega}}_i - 0.25 \left(\bar{\omega}_i^T \bar{\omega}_i\right) \mathbf{p}_i$, [2], $\ddot{\mathbf{q}}$ eventually becomes available. Thus, the cornerstone of the algorithm is the computation of λ, which is carried out iteratively as the solution of the *reduced system* $\mathbf{E}_1 \lambda = \eta \mathbf{M}^{-1} \mathbf{f} + \rho \bar{\mathbf{J}}^{-1} \hat{\mathbf{n}} - \tau$. Since the matrices $\mathbf{M}$ and $\bar{\mathbf{J}}$ are positive definite, the *reduced matrix* $\mathbf{E}_1 = \left(\eta \mathbf{M}^{-1} \eta^T + \rho \bar{\mathbf{J}}^{-1} \rho^T\right) \in \Re^{m \times m}$, with $\eta = \eta(\mathbf{\Phi},\mathbf{q})$, $\rho = \rho(\mathbf{\Phi},\mathbf{q})$, is also positive definite.

2. Preconditioning. A topology-based direct sparse solver

The preconditioning of the iterative solver for the reduced system is based on a direct solution of this system, in which the topology of the model is leveraged to efficiently compute $\mathbf{E}_1$, and to perform sparse, low fill-in factorization.

In what follows, two bodies b_i and b_k are called *j–adjacent* if they are connected through joint j. Since there is an ordering relationship among the bodies of a model, of the two adjacent bodies one has a lower index, and it is called the left–body, or `l`–body, while the higher–index body is called the `r`–body. They are denoted by $\mathtt{l}(j)$, and $\mathtt{r}(j)$, respectively. The *b_i–connectivity set* of body b_i is defined as the union of all joints that link body b_i to other bodies in the system, and it is denoted by $\zeta(b_i)$. The *joint*

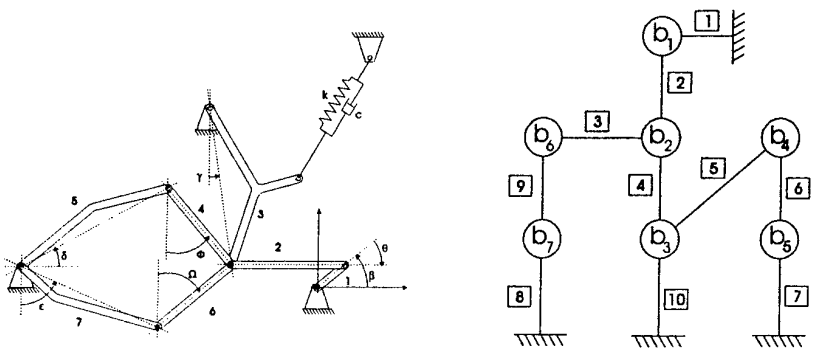
index $\mathcal{J}(b_i)$ of a body b_i is defined as the number of elements in $\zeta(b_i)$. The *topology index* $\mathcal{J}$ of a mechanism is defined as the largest joint index of any body in the mechanism. In this context, it is shown in [4] that an upper limit on the number of operations to compute $\mathbf{E}_1$ is $72\mathcal{J}(\mathcal{J}+1)\,n_b$ additions, and $126\mathcal{J}(\mathcal{J}+1)\,n_b$ multiplications. These numbers refer to block matrix operations, and the rule is that each joint leads to operations with block matrices of dimension equal to the number of constraint equations it induces. Thus, the largest dimension of any block matrix operation is 6×6, induced by a joint that removes all six relative degrees of freedom of a body. It follows that the number of operations only increases linearly with the number of bodies, and it is the topology index of the mechanism that, from a connectivity stand point, influences the computational effort.

The direct solution of the reduced system is obtained by repeatedly applying a two–stage process, [4], an approach similar to the one proposed in [3]. First, a Lagrange multiplier λ_j associated with joint j is solved for in its defining equation (the *isolation (I)* stage), and then eliminated from the defining equations of all joints $k \in \zeta(1(j)) \cup \zeta(r(j))$ (the *elimination (E)* stage). The two factors that influence the effort to compute the solution of the reduced system are the elimination order, and the topology index of the mechanism. The importance of the elimination order is illustrated in [4], where the direct solution of the reduced system associated with the Andrew's squeezing mechanism is analyzed in terms of block matrix additions (A), multiplications (M), inversions (I), and fill–in (F). The results are showed in Table 1, the mechanism and the associated topology graph are presented in Fig.1, in which bodies map into the graph's vertices, while the graph's edges correspond to physical joints. For the good elimination sequence 1 2 7 6 5 8 9 3 4 10, the number of additions and multiplications is reduced by roughly 70%, compared to the case when a bad elimination sequence is employed. Likewise, although this mechanism has closed loops, when using the good elimination, the algorithm results in no fill-in. It follows that rearranging the ordering of the joints during the preprocessing stage of the simulation results in increased solution efficiency at each integration step.

TABLE 1. Andrew's mechanism. Operation count.

Elimination Sequence	*A*	*M*	*I*	*F*
1 2 7 6 5 8 9 3 4 10	35	35	21	0
4 2 3 5 6 9 1 7 8 10	111	130	40	19

For a class of topology index 2 mechanisms; i.e., a chain of pendulums,

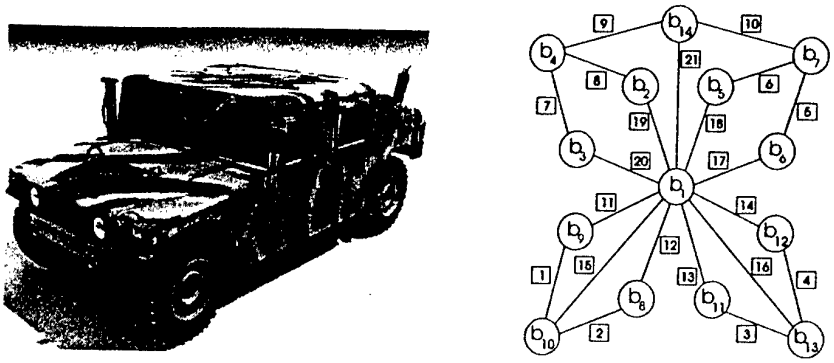


a. Model b. Topology graph

Figure 1. Seven body mechanism

it is shown in [4] that the reduced system is solved in $O(N_J)$ effort. However, as pointed out in [1], this ceases to be the case for star-like topologies; i.e., models with high topology index. These are models in which one body is connected to many other bodies in the model. In the limit, the performance of the two stage *isolation–elimination (IE)* algorithm turns out to be $O(N_J^3)$. The algorithm remains order $O(N_J)$ though, provided *topology index reduction* is first applied to the model, [4]. The topology index reduction amounts to a virtual break-up of the high index body into an appropriate number of smaller virtual bodies connected by fixed joints; i.e., joints that remove all relative degrees of freedom. This operation reduces the index of the mechanism while increasing the number of unknowns; i.e., Lagrange multipliers in the new reduced system. The new bodies and joints are called virtual because they do not have a physical counterpart. Their effect is a topology change for the sole purpose of leading to an equivalent but simpler reduced linear system. The idea behind topology index reduction is that a star-like topology should be regarded as the result of a bad elimination sequence applied to a virtual mechanism. The effort for computing the Lagrange multipliers is expected to decrease by going back to this virtual mechanism via topology index reduction, and then applying a better elimination sequence on its reduced matrix.

Topology index reduction was applied in [4] to star-like topologies with indexes anywhere from 3 to 16. The results indicate that even for this type of topology, the number of operations only increases linearly with the number of joints in the model, provided the topology index of each model is first reduced to 3 or 4. Topology index reduction was also applied to a High Mobility Multi-Wheeled Vehicle (HMMWV) in Fig.2, a model with topology index $\mathcal{J} = 11$. The impact of the elimination order and topology index reduction are presented in Table 2, where NNZ indicates



a. Picture b. Topology graph

Figure 2. HMMWV example

TABLE 2. HMMWV reduced system solution effort.

Elimination Sequence	A	M	I	F	NNZ
Bad	1240	1336	195	96	99
Good	459	469	109	10	99
Index reduction	220	233	90	13	77

the number of non-zero blocks in $\mathbf{E}_1$. The reader is referred to [4] for a detailed account of how the topology index reduction was done, and what were the "good" and "bad" elimination sequences used for the solution of the reduced system.

3. The iterative solution of the reduced system

If $\mathbf{W}$ is the preconditioning matrix, the algorithm *Preconditioned Conjugate Gradient* below guarantees a solution of the reduced system $\mathbf{E}_1\lambda = \mathbf{b}$ within m iterations. In this algorithm, the preconditioner is responsible for finding $\mathbf{c}_k$, an operation presented in Section 2 and based on a direct solution of the reduced system. In the context of parallelizing the iterative algorithm on a per body basis using a shared memory framework provided by the OpenMP standard, the tasks specific to the iterative solver are the computation of $\hat{\mathbf{e}}_k = \mathbf{E}_1\mathbf{d}_k \in \Re^m$ and $\hat{e}_k = \mathbf{d}_k^T\mathbf{E}_1\mathbf{d}_k \in \Re$, [4]. Defining the constraint index of body b_i as $\mathcal{C}(b_i) = \sum_{l\in\zeta(b_i)} m_l$, it is shown in [4] that the number of operations on the thread associated with body b_i during each iteration is $12\mathcal{C}(b_i)+12$ multiplications, and $11\mathcal{C}(b_i)-5\mathcal{J}(b_i)+5$ additions. This indicates that the computational effort per body and per iteration is

linear in the body constraint index, and leads to the conclusion that load balancing is obtained when the bodies in the model have identical or close constraint indexes $C(b_i)$. It follows that topology index reduction also helps the iterative solver by balancing the thread load, and distributing the work to more threads. Note that the performance of the iterative solver is not impacted at all by the existence of closed loops in the model.

Preconditioned Conjugate Gradient.

$k = 0; \lambda_k = \mathbf{0}; \mathbf{e}_k = \mathbf{b}$
while $(\mathbf{e}_k \neq \mathbf{0})$
  Solve $\mathbf{W}\mathbf{c}_k = \mathbf{e}_k$
  $k = k + 1$
  if $(k = 1)$
    $\mathbf{d}_1 = \mathbf{c}_0$
  else
    $\beta_k = \left(\mathbf{e}_{k-1}^T \mathbf{c}_{k-1}\right) / \left(\mathbf{e}_{k-2}^T \mathbf{c}_{k-2}\right)$
    $\mathbf{d}_k = \mathbf{e}_{k-1} + \beta_k \mathbf{d}_{k-1}$
  end
  $\alpha_k = \left(\mathbf{e}_{k-1}^T \mathbf{c}_{k-1}\right) / \left(\mathbf{d}_k^T \mathbf{E}_1 \mathbf{d}_k\right)$
  $\lambda_k = \lambda_{k-1} + \alpha_k \mathbf{d}_k$
  $\mathbf{e}_k = \mathbf{e}_{k-1} - \alpha_k \mathbf{E}_1 \mathbf{d}_k$
end
$\lambda = \lambda_k$

The multi-threaded attribute of the algorithm draws upon the mapping of each body on a simulation thread, and it is the cornerstone of the proposed solution method. Each body-thread starts with the computation of specific kinetic and kinematic quantities, and continues through the numerical solution; i.e., through the iterative solvers and numerical integration. Additional implementation details, and a discussion on how the iterative approach is used in the framework of SSODE integration to compute the dependent position and velocity by means of a different reduced matrix $\mathbf{E}_2$ is provided in [4].

References

1. Baraff, D.: 1996,'Linear-Time Dynamics using Lagrange Multipliers', *COMPUTER GRAPHICS Proc., Annual Conference Series*, pp.137–146.
2. Haug, E.J.:*Computer Aided Kinematics and Dynamics of Mechanical Systems.* Allyn and Bacon, Boston, London, Sydney, Toronto, 1989.
3. Lubich, C. U. Nowak, U. Pöhle, and C. Engstler: 1992 MEXX–Numerical software for the integration of constrained mechanical multibody systems, *Techical Report SC 92–12, ZIB Berlin, Germany.*
4. Negrut, D.: 2002,'Linear algebra considerations for the multi-threaded simulation of mechanical systems represented in Cartesian coordinates', submitted, *Multibody System Dynamics.*
5. Wehage, R. and E. Haug: 1982,'Generalized Coordinate Partitioning for Dimension Reduction in Analysis of Constrained Dynamic Systems', *ASME J. Mech. Design*, Vol.104 no.1, pp.247–255.

Symofros: A Virtual Environment for Modeling, Simulation and Real-Time Implementation of Multibody System Dynamics and Control

Jean-Claude Piedbœuf, József Kövecses, Brian Moore[†], Régent L'Archevêque
Space Technologies, Canadian Space Agency, 6767 Route de l'Aéroport, St-Hubert, Québec, Canada, J3Y 8Y9
Jean-Claude.Piedboeuf@space.gc.ca, Jozsef.Kovecses@space.gc.ca, Brian.Moore@space.gc.ca, Regent.Larcheveque@space.gc.ca

Abstract. This paper briefly describes Symofros, the modeling, simulation and control environment developed and used at the Canadian Space Agency for multibody and robotic systems. This environment is based on a symbolic modeling and code generation engine supported by Maple, and the Matlab/Simulink environment. Symofros serves two main purposes: control and real-time implementation, and analysis and design. Applications of the Symofros environment in space robotics will also be demonstrated in this paper.

1. Introduction

Multibody dynamics is of central importance in design and analysis of mechanical systems and their controllers. In space systems, multibody modeling and analysis is the fundamental element in developing and operating systems and technologies. Simulations (both non-real-time and real-time) are required for space robotics and space systems in general. The Canadian Space Agency's (CSA) in-house multibody dynamics software package Symofros has been developed since 1994. Symofros permits modeling, simulation and real-time control of multibody systems. The software architecture of Symofros is based on the Maple symbolic modeling engine and the Matlab-Simulink environment. Symofros is used for various projects in robotics both inside and outside CSA.
This paper describes the integrated virtual environment provided by Symofros. This environment allows the user to efficiently model, simulate in non-real-time and in real-time, and then do the implementation on a real hardware. The paper details the modeling environment based on XML, Maple and on a server system. We will then discuss the generation of the functions used for the simulation and the controller development. We will describe how a system can be simulated using the libraries built in Symofros. The next stage is the generation of a real-time simulation. As it will be discussed in the following,

[†] Opal-RT Technologies on secondment to Canadian Space Agency

the Symofros architecture provides a very flexible environment that allows users to perform rapid prototyping. For example, the user can test, in the real-time environment, a complex model-based controller using a model of a robot, and then by simply clicking on a button, switch to the control of the real-hardware.

2. Modeling

Symofros multibody dynamics engine is based on a formulation relying on Jourdain's principle. Jourdain's principle provides a physically clear framework for multibody analysis for both holonomic and nonholonomic systems. Jourdain's principle, as a differential variational principle, possesses two very important features. It is invariant under transformations from one possible set of coordinates to another, and expresses the main principle of constrained systems, that the virtual power of constraint forces for admissible velocity variations (or virtual velocities in other words) vanishes. These are the two fundamental elements upon which the analysis of complex systems can be based. The various parts of Symofros' modeling engine have been extensively validated by experiments, analytical examples and simulations.

Complex systems (e.g. closed-loop multibody systems, parallel robots) can be split to sub-systems, and the system model can then be assembled by employing constraints between the various sub-systems. Open-loop systems and sub-systems are modeled using a generic recursive formulation, which can consider both rigid and flexible elements in the system (Piedbœuf, 1998). In general, the consideration of the system constraints is a key issue in multibody dynamics. Symofros is able to handle both holonomic and nonholonomic constraints based on the Lagrangian multiplier technique with Baumgarte stabilization, and the use of projection and decomposition techniques. Work is in progress to develop new advanced methods for handling constrained system dynamics, and to extend the capabilities and include various new and existing approaches in the simulation environment.

The Symofros environment is able to consider rigid and flexible bodies as elements of a multibody system. Currently, flexible beams are implemented for flexible body modeling with various choices of shape functions. Besides the traditional assumed modes approximations, a characteristic modeling approach employed is the advanced use of the assumed modes method, where the discretization is carried out in a way similar to the finite element method, i.e. interpolation functions are generated locally for an element, but then the shape functions are represented globally as in the traditional assumed modes method. Flexible plate models are planned to be included in the near future. Besides body flexibility, the finite stiffness of the mechanical structure of the

connecting joints is also a dominant effect in multibody systems. Symofros is capable of modeling joint flexibility using discrete stiffness models.
For contact mechanics modeling, Symofros currently uses the Contact Dynamics Toolkit developed by MacDonald Dettwiler Space and Advanced Robotics Ltd. Work is in progress to extend the contact-impact modeling capabilites of the Symofros environment with special attention to the real-time aspects. There are two main approaches being investigated in contact dynamics modeling: the local compliance based models, and the rigid body models based on unilateral constraints.
Dynamic parameter identification is an important area in multibody systems simulations, analysis and control. This area is currently being looked at to develop an identification toolbox for Symofros The two main purposes of the identification toolbox is to facilitate the optimum generation of experimental data for identification, and to process the measured data to determine the required parameters. This work involves the formulation and analysis of the dynamic equations in the form suitable for identification, and the solution techniques of these equations for the parameters.
Symofros also includes a control system toolbox comprising a library of Simulink blocks of various control algorithms (e.g. model based control with PD compensation). These can be easily linked and tested with the dynamic model of a multibody system to form the model of a controlled system. Also, new control algorithms can be readily built from the existing primitives.

3. Symofros software architecture

Symofros is based on commercial tools and is composed of three main modules for mechanical system description, modeling and simulation (see Figure 1).
Creating a model of a mechanical system consists of describing the bodies, the joints and the topology of the system. This model description is based on the XML language[1], a standardized language used to describe any kind of data and used for many applications. For mechanical system description, this language is also used by researchers in Spain (Rodríguez et al., 2001).
The Symbolic Model Generator (SMG) comprises modules written in the Maple language to perform the symbolic modeling. The input of the module is an XML or Maple file describing the properties of the mechanical system. This file is used by the module to compute the kinematic and dynamic quantities of the bodies and the joints. From the input file, the topology of the mechanical system is analyzed to generate a graph model. Using the topology with the body and joint data, the SMG develops the kinematic equa-

[1] www.w3.org

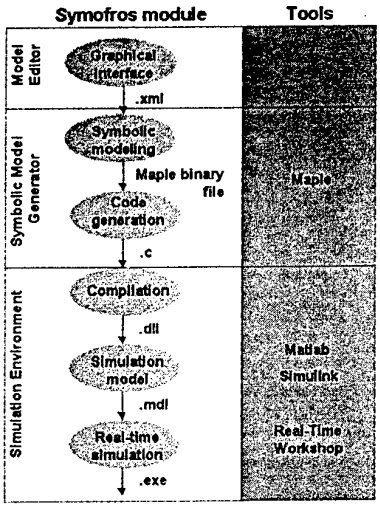


Figure 1. Overview of Symofros modules

tions. Using the kinematic formulation, the SMG builds the dynamic equations in various forms, for simulation (forward dynamics), control (inverse dynamics), and parameter identification (currently in development). Special kinematic quantities are also generated for parallel mechanisms based on the approach proposed in (Monsarrat and Gosselin, 2002). The SMG is normally used as an automatic model generator, but it is also a powerful tool to analyze the dynamic equations and to develop models on-line. More details on the symbolic modeling part of Symofros can be found in (Piedbœuf, 1996) and (Moore et al., 2002).

For simulation and real-time implementation, the SMG generates C code to represent the multibody system. The code generation requires optimization tools to break the complex expressions down to smaller expressions. This also helps improving the code efficiency for simulation since sub-expressions appearing several times need to be computed only once. The C functions are the links between the modeling part of Symofros, and the simulation/real-time implementation parts. Therefore, using the model in an advanced simulation or in the real-time environment is straightforward.

The Symofros SMG module can also be called using a server. The user has to connect to the server and send the mechanical system description files. These files are then processed by the SMG and the C file and processing information are sent back to the user. This approach helps protecting the Symofros source code, which is located on the Maple server and not accessible by the user. This also reduces the maintenance required, since the upgrades and modifications have to be carried out on the server only. Also, using the server reduces the load on the user's computer ressources.

To allow an efficient and convenient use of the mathematical model derived, and to enable the numerical simulation, Symofros is directly linked to the Matlab/Simulink environment. The Simulink environment allows to create complex models and generate complex simulation systems in only a few simple steps without the need of advanced programming skills. Special blocks are available in the library in order to call the functions generated symbolically and written in the .c file. As an example, Figure 2 shows how the forward dynamics can be computed. In this example, the dark blocks (*Mnl*, *gnl*) are used to call the functions written in the .c file. Then, using standard Simulink blocks, the system of equations is solved to obtain the accelerations, and integrated to obtain the generalized velocities and generalized coordinates. This block (Forward Dynamics) can then be found in the Symofros library and re-used with other models.

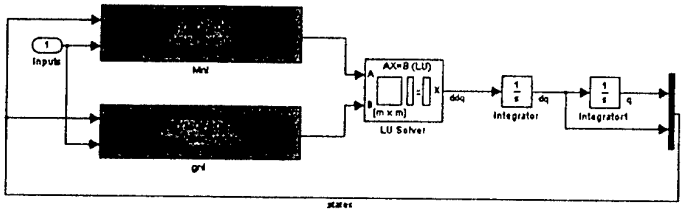


Figure 2. Simulation model within Simulink

Real-time simulation and hardware-in-the-loop simulation can be achieved by using complementary tools like the Real-Time Workshop and RT-Lab for generating real-time simulation code and distributing the computations on several computers. More details on this topic can be found in the next section and in (L'Archevêque et al., 2000), (Lambert et al., 2001) and (Piedbœuf et al., 2001).

4. Applications

Canada's contribution to the International Space Station (ISS) is the Mobile Servicing System (MSS) which is composed of the Mobile Transporter, the Space Station Remote Manipulator (SSRMS) and the Special Purpose Dexterous Manipulator (SPDM). The SPDM will be used to manipulate Orbital Replacement Units (ORUs) or scientific payloads. An important aspect of a typical SPDM task is the insertion/extraction of payloads. To support the MSS, CSA has developed the STVF[2] and the SMP[3]. Both of these systems

[2] SPDM Task Verification Facility

[3] System for Maintaining, Monitoring MRO Performance on board the ISS

are based on the Symofros engine. These two systems also demonstrate the two main application areas of Symofros: model based control, and simulation and analysis.

4.1. STVF: HARDWARE-IN-THE LOOP SIMULATOR

Due to the complexity of an SPDM task, a verification of the operation must be performed on the ground for each ORU manipulation. The main difficulty in this validation is verifying the part of the task for which the SPDM end-effector or payload undergoes contact with the environment. This part is verified using a hardware-in-the-loop simulation (HLS) to generate the real contact force using a mockup of the payload that needs to be manipulated.
The STVF Manipulator Testbed (SMT) (Aghili et al., 1999) is used to perform the HLS. The output of a real-time simulator representing a space robot is used as the input to the ground robot controller. The real contact forces are measured and fed back to the simulator. This approach is very flexible since we can represent not only SPDM but also other space manipulators.
Figure 3 shows the hardware architecture required for the test-bed. The real-time simulation is performed using the MSS Operation and Training Simulator (MOTS). The simulator includes the dynamics of the mobile base and the SSRMS in addition to the two arms of the SPDM. The full model has more than 50 degrees of freedom. The dynamic engine (SMT-SIM) is running at 1 kHz on an Origin 200 machine with four processors. The visualisation is running at 25 Hz on a four processors ONYX machine. The real-time control of the robot is achieved using a cluster of Pentium processors running QNX, and using Simulink Real-Time Workshop with Opal-RT RT-LAB for the code generation and multi CPU management. The graphical user interface on the SMT-CS is developed using Labview. The communication between Labview and the real-time system is managed by RT-LAB. The models required for the

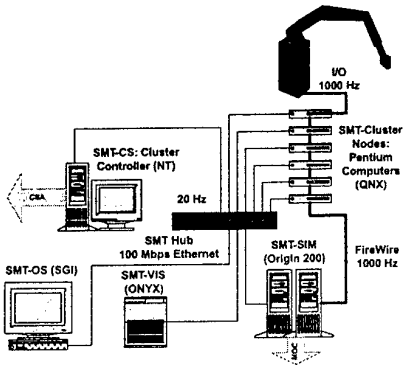


Figure 3. Computer Architecture for the HLS

controller and for the simulation on the cluster are generated using Symofros. The robot controller in the HLS mode is based on a cartesian feedback linearisation (de Carufel et al., 2000). For the design and the tune-up phases, we developed an equivalent model of the robot using Symofros. This model reproduces exactly the same interfaces (in terms of inputs and outputs) as the SMT robot has. Therefore, we can choose between the real robot and the simulated robot simply by clicking a switch. For the same reason, a simplified model with a reduced number of degrees of freedom has been developed for the space robot using Symofros. This model uses exactly the same interface as SMT-SIM. This flexibility is critical for the development since this allows the engineers to develop the overall software architecture in their offices and then download the code on the real-time system. There is no re-coding necessary between the pure simulation phase and the HLS phase.

4.2. SMP SIMULATOR

Experimental tests and analysis have shown that the capture of free-flyers is the most complicated task to be performed by a robotic operator on board of the ISS. The understanding of SSRMS and free-flyer dynamics require highly qualified and well-trained operators. The dexterity and accuracy of the astronauts may decrease over time if they are not trained on-board. It was an obvious choice to have a simulator on-orbit to keep the skills of the astronauts at the required level. In order to support the training scenarios required by the on-orbit training, the SMP [4] simulator has been developed. The main objective of the simulator is to determine if an astronaut is ready to perform an operation with the real SSRMS. The training scenario, implemented in the SMP, consists of capturing a free-flyer with the SSRMS.

The simulator is composed of four modules, the Graphical User Interface (GUI), the Analysis Module, the Visual Renderer (VR) and the Dynamic Simulator (SIM). It has the same architecture as the Basic Operations Robotic Instructional System (BORIS) simulator used to provide generic robotic training to the astronauts (L'Archevêque et al., 2001).

The GUI has been developed with Labview 6 and runs on Windows operating systems. During a training session, the operator is firstly prompted to log into the system. Then, he has the choice to start a simulation session, a session analysis or a trend analysis. The session analysis provides information such as the hand-controller rates, the relative position and velocity between the end-effector and the free-flyer, and the capture status. Operational criteria and heuristics are used to provide a score, which allows the astronaut to have a good picture of his personal progress over time using trend analysis. The astronaut can then determine if he needs more training or not. Figure 4 shows

[4] System for Maintaining, Monitoring MRO Performance on board the ISS

the VR model of a generic free-flyer as viewed by the SSRMS end-effector camera. The VR module has been developed with OpenGL toolboxes.
Two Symofros models have been used to represent the SSRMS and the free-flyer. The SSRMS model has been configured and tuned using real flight data (data gathered during SSRMS operations) in order to obtain a realistic model. Generic parameters have been established to configure the dynamic behavior of the free-flyer. A Simulink diagram, using Symofros toolboxes, performs the simulation of the SSRMS, models the attitude control system of the free-flyer, interprets the hand-controller input values, handles the capture sequence, and gathers session data. The SMP simulator running in soft real-time on Windows 2000 has been generated using Real-Time Workshop. The experimental system will be launched in January 2003 and will be used by several astronauts and cosmonauts.

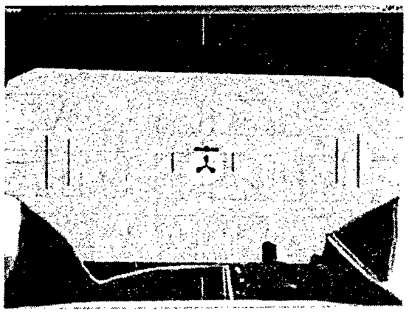


Figure 4. SMP Visual Renderer

References

Aghili, F., E. Dupuis, J.-C. Piedbœuf, and J. de Carufel: 1999, 'Hardware-in-the-Loop Simulations of Robots Performing Contact Tasks'. In: M. Perry (ed.): *Fifth International Symposium on Artificial Intelligence, Robotics and Automation in Space.* Noordwijk, The Netherland, pp. 583–588, ESA Publication Division.

de Carufel, J., E. Martin, and J.-C. Piedbœuf: 2000, 'Control Strategies for Hardware-in-the-Loop Simulation of Flexible Space Robots'. *IEE Proceedings-D: Control Theory and Applications* **147**(6), 569–579.

Lambert, M., B. Moore, and M. Ahmadi: 2001, 'Essential Real-Time and Modeling Tools for Robot Rapid Prototyping'. In: *The 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: i-SAIRAS 2001.*

L'Archevêque, R., M. Doyon, J.-C. Piedbœuf, and Y. Gonthier: 2000, 'SYMOFROS: Sofware Architecture and Real Time Issues'. In: *DASIA 2000 - Data Systems in Aerospace.* Montreal, Canada.

L'Archevêque, R., Z. Joukakelian, and P. Allard: 2001, 'BORIS: A Simulator for Generic Robotic Training'. In: *The 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: i-SAIRAS 2001.*

Monsarrat, B. and C. Gosselin: 2002, 'Jacobian Matrix of General Parallel and Hybrid Mechanisms with Rigid and Flexible Links: A Software-Oriented Approach'. In: *Proceedings of the 2002 ASME Design Engineering Technical Conferences*.

Moore, B., J.-C. Piedbœuf, and L. Bernardin: 2002, 'Maple as an automatic code generator ?'. In: *Maple Summer Workshop*.

Piedbœuf, J.-C.: 1996, 'Modelling Flexible Robots with Maple'. *Maple Tech: The Maple Technical Newsletter* **3**(1), 38–47.

Piedbœuf, J.-C.: 1998, 'Recursive Modelling of Flexible Manipulators'. *The Journal of Astronautical Sciences* **46**(1).

Piedbœuf, J.-C., F. Aghili, M. Doyon, Y. Gonthier, E. Martin, and W.-H. Zhu: 2001, 'Emulation of Space Robot Through Hardware-in-the-Loop Simulation'. In: *The 6th International Symposium on Artificial Intelligence and Robotics & Automation in Space: i-SAIRAS 2001*. Canadian Space Agency, St-Hubert, Quebec, Canada.

Rodríguez, J. I., J. M. Jimńez, F. J. Funes, and J. Garcià de Jalón: 2001, 'Dynamic Simulation of Multi-Body Systems on Internet using Corba, Java and XML'. In: *USACM, Sixth U.S. National Congress on Computational Mechanics*. p. 398.

ON CALCULATION OF JACOBIAN MATRICES IN SIMULATION OF MULTIBODY SYSTEMS*

D.Yu. POGORELOV
Bryansk State Technical University
b. 50 let Oktyabrya 7, 241035 Bryansk, Russia

1. Introduction

Integration of stiff equations of motion of multibody systems using implicit numerical methods, calculation of equilibrium positions, linearization of equations, constructing optimal controls and some other important tasks require computations of Jacobian matrices. Evaluations of the matrices by finite differences in the case of stiff equations is about 13 times more expensive than that for the mass matrix of the system [1]. Decreasing the corresponding computational efforts could improve the efficiency of numerical analysis of large multibody systems. In this paper the analytic expressions for Jacobian matrices are derived. The method of derivation is based on the composite rigid body algorithm [2], which allows considerable reducing computational efforts in evaluations of the matrices. Multibody systems with holonomic ideal stationary constraints are considered. If a system has closed loops, a minimal number of joints must be cut. An ordered numeration 1...n of bodies and joints is introduced. Let us consider a chain of the system tree, which begins at body 0 (the inertial frame). Indices of bodies increase along the chain, i.e. the minimal index has the body connected with body 0. Indices of joints in the chain are equal to those of bodies, namely joint j connects bodies i and j, $j > i = j^-$. The following important sets are used below: $J(k)$ is the set of indices of joints included in the path from body k to body 0; $B(k)$ is the set of indices of bodies, so that their paths to body 0 contain joint k. Let $q = (q_1^T \ \cdots \ q_n^T)^T$ be coordinates of the system, q_j is the $n_j \times 1$ matrix of local coordinates in joint j, which specifies the position of body j relative to body j^-.

2. Jacobian matrices of kinematic variables

Let $r_j, v_j, a_j, A_{0j}, \omega_j, \varepsilon_j$ be the radius vector, velocity and acceleration of the origin as

* Supported by the RFBR under grant 02-01-00364 and by the scientific program "Universities of Russia – basic research" UR.04.01.046

well as the direction cosine matrix, angular velocity and angular acceleration of a body-fixed frame. The following recursive relations are valid:

$$r_j(q) = r_i(q) + r_j^r(q_j), \quad A_{0j}(q) = A_{0i}(q)A_j^r(q_j),$$

$$v_j = v_i + \tilde{\omega}_i r_j^r + v_j^r, \quad \omega_j = \omega_i + \omega_j^r,$$

$$a_j = a_i + \tilde{\varepsilon}_i r_j^r + \tilde{\omega}_i\tilde{\omega}_i r_j^r + 2\tilde{\omega}_i v_j^r + a_j^r, \quad \varepsilon_j = \varepsilon_i + \tilde{\omega}_i\omega_j^r + \varepsilon_j^r,$$

where $r_j^r, A_j^r, v_j^r, \omega_j^r, a_j^r, \varepsilon_j^r$ specify the position and motion of body j relative to body i,

$$v_j^r = d_j\dot{q}_j, \quad \omega_j^r = b_j\dot{q}_j,$$

$$a_j^r = d_j\ddot{q}_j + \alpha_j, \quad \varepsilon_j^r = b_j\ddot{q}_j + \beta_j.$$

Introducing the global velocities and accelerations

$$V_j = (\, v_j^T \quad \omega_j^T)^T, \quad W_j = (\, a_j^T \quad \varepsilon_j^T)^T$$

yields the recursive formulas

$$V_j = C_{ij}V_i + S_j\dot{q}_j, \quad W_j = C_{ij}W_i + S_j\ddot{q}_j + \Gamma_j^r,$$

$$C_{ij} = \begin{pmatrix} I_3 & -\tilde{r}_j^r \\ 0 & I_3 \end{pmatrix}, \quad S_j = \begin{pmatrix} d_j \\ b_j \end{pmatrix}, \quad \Gamma_j = \begin{pmatrix} \alpha_j \\ \beta_j \end{pmatrix},$$

and the explicit expressions

$$V_j = \Phi_j\dot{q}, \quad W_j = \Phi_j\ddot{q} + \Psi_j,$$

$$\Phi_j = C_{0j}\sum_{k\in J(j)}\hat{S}_k^*.$$

Here we introduced the notations, which we use throughout the paper for $6\times n_j$ matrices X_k: $X_k^* = C_{0k}^{-1}X_k$, as well as for $m\times n_j$ matrices:

$$\hat{X}_k = (0 \quad \cdots \quad 0 \quad X_k \quad 0 \quad \cdots \quad 0).$$

To derive the explicit Jacobian matrices for the kinematic variables with respect to coordinates q, the variations of coordinates δq are used. Let $\delta r_j, \delta\pi_j$ be the corresponding displacement of the origin and the rotation vector of the body-fixed frame, $\delta\tilde{\pi}_j = \delta A_{0j}A_{j0}$:

$$\delta R_j = (\delta r_j^T \quad \delta\pi_j^T)^T = \Phi_j\delta q.$$

Variations of recursive relations for velocities and accelerations yield:

$$\delta V_j = C_{ij}\delta V_i - \Pi_j^v\delta\pi_i + S_j^v\delta q_j, \quad \delta W_j = C_{ij}\delta W_i - \Omega_j\delta\omega_i - \Pi_j^a\delta\pi_i + S_j^a\delta q_j,$$

$$\Pi_j^v = \begin{pmatrix} \tilde{\omega}_i\tilde{r}_j^r + \tilde{v}_j^r \\ \tilde{\omega}_j^r \end{pmatrix}, \quad S_j^v = \begin{pmatrix} d_j^v \\ b_j^\omega \end{pmatrix} = \begin{pmatrix} \tilde{\omega}_i d_j + v_j'^r \\ \omega_j'^r \end{pmatrix},$$

$$\Omega_j = \begin{pmatrix} 2r_j^r\omega_i^T + 2\tilde{v}_j^r - \omega_i r_j^{rT} - \omega_i^T r_j^r I_3 \\ \tilde{\omega}_j^r \end{pmatrix},$$

$$\Pi_j^a = \begin{pmatrix} (\tilde{\omega}_i\tilde{\omega}_i + \tilde{\varepsilon}_i)\tilde{r}_j^r + 2\tilde{\omega}_i\tilde{v}_j^r + \tilde{a}_j^r \\ \tilde{\omega}_i\tilde{\omega}_j^r + \tilde{\varepsilon}_j^r \end{pmatrix}, \quad S_j^a = \begin{pmatrix} (\tilde{\omega}_i\tilde{\omega}_i + \tilde{\varepsilon}_i)d_j + 2\tilde{\omega}_i v_j'^r + a_{qj}'^r \\ \tilde{\omega}_i\omega_j'^r + \varepsilon_{qj}'^r \end{pmatrix},$$

$$v_j'^r = \partial v_j^r / \partial q_j^T \ , \ \omega_j'^r = \partial\omega_j^r / \partial q_j^T \ , \ a_{qj}'^r = \partial a_j^r / \partial q_j^T \ , \ \varepsilon_{qj}'^r = \partial\varepsilon_j^r / \partial q_j^T \ .$$

Here I_3 is the 3×3 identity matrix.

Thus, the Jacobian matrices result from the following expressions:

$$\delta V_j = C_{0j} \sum_{k\in J(j)} (\hat{S}_k^{v*} - \mathbf{\Pi}_{jk}^v \hat{b}_k)\,\delta q \,,$$

$$\delta W_j = C_{0j} \sum_{k\in J(j)} \left(\hat{S}_k^{w*} - \mathbf{\Omega}_{jk}\hat{b}_k^\omega + (\mathbf{\Omega}_{jk}\boldsymbol{\pi}_k^\omega - \mathbf{\Pi}_{jk}^a)\,\hat{b}_k\right)\delta q \,,$$

where

$$\mathbf{\Pi}_{jk}^v = \mathbf{\Pi}_j^v - \mathbf{\Pi}_k^v, \qquad \mathbf{\Pi}_j^v = (\boldsymbol{\pi}_j^{vT} \quad \boldsymbol{\pi}_j^{\omega T})^T = \sum_{m\in J(j)} \Pi_m^{v*} \,,$$

$$\mathbf{\Pi}_{jk}^a = \mathbf{\Pi}_j^a - \mathbf{\Pi}_k^a, \qquad \mathbf{\Pi}_j^a = \sum_{m\in J(j)} (\Pi_m^{a*} - \Omega_m^* \boldsymbol{\pi}_m^\omega) \,,$$

$$\mathbf{\Omega}_{jk} = \mathbf{\Omega}_j - \mathbf{\Omega}_k, \qquad \mathbf{\Omega}_j = \sum_{m\in J(j)} \Omega_m^* \,.$$

The next formulas are the variation of the matrix S_j^*

$$\delta S_j^* = \delta\left(C_{0j}^{-1} S_j\right) = \delta' S_j^* + \begin{pmatrix} \delta\tilde{\pi}_j & (\delta\tilde{r}_j + \tilde{r}_j\delta\pi_j)^{\sim} \\ 0 & \delta\tilde{\pi}_j \end{pmatrix} S_j^* ,$$

$$\delta' S_j^* = C_{0j}^{-1} \begin{pmatrix} A_{0j}\delta d^j \\ A_{0j}\delta b^j \end{pmatrix} = \sum_{m=1}^{n_j} S_{jm}'^* \delta q_{jm}, \quad \delta q_j = (\delta q_{j1} \ \cdots \ \delta q_{jn_j})^T \,,$$

and the important variation $\delta(\Phi_j^T X)$ for an arbitrary 6×1 matrix X

$$\delta(\Phi_j^T X) = \sum_{k\in J(j)} \hat{S}'_{Xk}\delta q + \sum_{k\in J(j)} \hat{S}_k^{*T} \left(\sum_{m\in J(k)} \tilde{X}_{\pi j} \hat{S}_m^* + \sum_{m\in J(j)-J(k)} \tilde{X}_{rj} \hat{S}_m^* \right) \delta q$$
$$+ \sum_{k\in J(j)} \hat{S}_k^{*T} C_{0j}^T \delta X .$$

The first summand in this formula contains the matrices $\hat{S}'_{Xk}$ with one diagonal $n_k \times n_k$ block $S'_{Xk} = (S_{k1}'^{*T} X^* \ \cdots \ S_{kn_k}'^{*T} X^*)$, the second one includes the matrices

$$X^* = C_{0j}^T X = \begin{pmatrix} x_r \\ x_\pi \end{pmatrix}, \quad \tilde{X}_{rj} = \begin{pmatrix} 0 & 0 \\ -\tilde{x}_r & \tilde{x}_r\tilde{r}_j \end{pmatrix}, \quad \tilde{X}_{\pi j} = \begin{pmatrix} 0 & \tilde{x}_r \\ 0 & \tilde{x}_r\tilde{r}_j + \tilde{x}_\pi \end{pmatrix}.$$

Introducing the variation $\delta\dot{q}$ allows deriving the Jacobian matrices with respect to $\dot{q}$. The following expressions specify the necessary matrices:

$$\delta V_j = \Phi_j \delta\dot{q}\,, \quad \delta W_j = C_{0j} \sum_{k\in J(j)} (\hat{S}_k^{w*} - \boldsymbol{\Omega}_{jk}\hat{b}_k)\,\delta\dot{q}\,,$$

$$S_j^w = \begin{pmatrix} 2\tilde{\omega}_i d_j + \partial a_j^r / \partial \dot{q}_j^T \\ \tilde{\omega}_i b_j + \partial \varepsilon_j^r / \partial \dot{q}_j^T \end{pmatrix}.$$

3. Jacobian matrices for equations of motion

Here we consider equations of motion of a tree–structured multibody system, derived with the help of the Newton-Euler formalism:

$$f(\ddot{q},\dot{q},q,t) = \sum_{j=1}^{n} \Phi_j^T(q) G_j(q,\dot{q},\ddot{q},t) = 0\,, \tag{1}$$

$$G_j = M_j(q) W_j(q,\dot{q},\ddot{q}) + k_j(q,\dot{q}) - f_j(q,\dot{q},t)\,,$$

$$M_j = \begin{pmatrix} m_j I_3 & 0 \\ 0 & J_j \end{pmatrix},\ k_j = \begin{pmatrix} 0 \\ \tilde{\omega}_j J_j \omega_j \end{pmatrix},$$

where m_j, J_j is the mass and the inertia tensor of body j, the 6×1 matrix f_j contains applied forces and moments reduced to the centre of mass. Origins of the body-fixed frames are located in the corresponding centres of mass.

Variation of equations with respect to q and $\dot{q}$ produces the Jacobian matrices

$$\delta f = J^q \delta q + J^v \delta\dot{q}\,.$$

Application of the results of the previous section as well as rearrangement of summations

$$\sum_{j=1}^{n} \sum_{k\in J(j)} = \sum_{k=1}^{n} \sum_{j\in B(k)}\,, \qquad \sum_{j=1}^{n} \sum_{k\in J(j)} \sum_{m\in J(j)} = \sum_{k=1}^{n} \sum_{m=1}^{n} \sum_{j\in B(\max(k,m))}$$

lead to the explicit expression for a separate block of the matrix J^v

$$J_{km}^v = S_k^{*T}\left(\mathbf{I}_u\left(S_m^{w*} + \boldsymbol{\Omega}_m b_m\right) - \left(\mathbf{I}_{\Omega u} + \begin{pmatrix} 0 & \mathbf{k}_{u\omega}'^T \end{pmatrix}^T\right) b_m + \mathbf{F}_{Vkm} S_m^*\right),\ u = \max(k,m)$$

with the following composite matrices:

$$\mathbf{I}_u = \sum_{j\in B(u)} M_j^*, \quad \mathbf{I}_{\Omega u} = \sum_{j\in B(u)} M_j^* \boldsymbol{\Omega}_j, \quad M_j^* = C_{0j}^T M_j C_{0j}\,,$$

$$\mathbf{k}_{u\omega}' = \sum_{j\in B(u)} k_{j\omega}'\,, \quad k_{j\omega}' = \tilde{\omega}_j J_j - (J_j\omega_j)^{\sim}\,,$$

$$\mathbf{F}_{Vkm} = \sum_{j\in B(k)} \sum_{i\in B(m)} C_{0j}^T f_{jVi}' C_{0i}^T\,, \quad f_{jVi}' = (\partial f_j / \partial v_i^T \quad \partial f_j / \partial \omega_i^T)\,.$$

The expression for the matrix J^q is more cumbrous:

$$J_{km}^q = S_{Gk}'\big|_{k=m} + S_k^{*T}(-\mathbf{F}_{Rkm} + \tilde{\mathbf{G}}_\pi\big|_{m\in J(k)} + \tilde{\mathbf{G}}_r\big|_{m\in B(k)-\{k\}}) S_m^* \tag{2}$$

$$+S_k^{*T}(((0 \quad \mathbf{k}'^T_{u\pi\omega})^T - \mathbf{I}_{\Pi u} + \mathbf{F}_{\Pi Vkm})b_m + \mathbf{I}_u(S_m^{w*} + \boldsymbol{\Omega}_m b_{m\pi\omega} + \boldsymbol{\Pi}_m^a b_m))$$

$$+S_k^{*T}(((0 \quad \mathbf{k}'^T_{u\omega})^T - \mathbf{I}_{\Omega u})b_{m\pi\omega} - \mathbf{F}_{Vkm}(S_m^{v*} + \boldsymbol{\Pi}_m^v b_m)),\ u = \max(k,m),$$

and includes the following composite and auxiliary matrices:

$$S'_{Gk} = (S'^{*T}_{k1}\mathbf{G}_k \quad \cdots \quad S'^{*T}_{kn_k}\mathbf{G}_k), \quad \mathbf{G}_k = \sum_{j\in B(k)} G_j^*,$$

$$\mathbf{F}_{\Pi Vkm} = \sum_{l\in B(m)}\sum_{j\in B(k)} C_{0j}^T f'_{jVl} C_{0l} \boldsymbol{\Pi}_l^v, \quad \mathbf{F}_{Rkm} = \sum_{j\in B(k)}\sum_{l\in B(m)} C_{0j}^T f'_{jRl} C_{0l},$$

$$\sum_{j=1}^{n} \Phi_j^T f_j + \sum_{s=n+1}^{m+n} (\Phi_v^T - \Phi_u^T C_{uv}^T)\lambda_s = 0, \quad \mathbf{k}'_{u\pi\omega} = \sum_{j\in B(u)} \left(k'_{j\pi} - k'_{j\omega}\boldsymbol{\pi}_j^{\omega}\right),$$

$$\mathbf{k}'_{u\omega} = \sum_{j\in B(u)} k'_{u\omega}, \quad k'_{j\pi} = J_j\tilde{\varepsilon}_j - (J_j\varepsilon_j)^{\sim} + \tilde{\omega}_j(J_j\tilde{\omega}_j - (J_j\omega_j)^{\sim}).$$

If we neglect the expenses for computing the composite matrices $\mathbf{F}_{Rkm}, \mathbf{F}_{Vkm}$ (most of the f'_{jVl}, f'_{jRl} matrices are usually zeroes), the evaluation of the matrices J^v, J^q for a chain requires $9n^2 + O(n)$ and $13.5n^2 + O(n)$ multiplications, $7.5n^2 + O(n)$ and $11n^2 + O(n)$ additions, respectively.

4. Jacobian matrices for equilibrium equations

Calculation of equilibrium positions and subsequent linearization of equations of motion is an important part of analysis of many technical multibody systems.

Consider a multibody system with m closed kinematic loops. If all cut joints are kinematic pairs with rotational and/or translational degrees of freedom (from 0 to 5 d.o.f.), the nonlinear equilibrium equations have the following form:

$$\sum_{j=1}^{n} \Phi_j^T f_j + \sum_{s=n+1}^{m+n} (\Phi_v^T - \Phi_u^T C_{uv}^T)\lambda_s = 0,$$

$$S_s^T \lambda_s = 0,\ s = n+1,\ldots,n+m,$$

$$r_v(q) - r_u(q) - A_{0u}(q) r_{uv}^u(q_s) = 0,\ -\frac{1}{2}\sum_{k=1}^{3} \tilde{e}_k A_{0u}(q) A_{uv}(q_s) A_{0v}^T(q)\, e_k = 0.$$

The last two equations correspond to closure conditions for cut joint s connecting bodies u,v, e_k are basis vectors of inertial frame; the 6×1 vector λ_s contains a reaction force and moment reduced to the centre of mass of body v.

The Jacobian matrix for the constraint equations is $\Phi_v - C_{uv}\Phi_u - \hat{S}_s$. To obtain the Jacobian matrix for the first equation with respect to coordinates, reactions should be added to applied forces

$$F_j = f_j - \sum_{s\in C_1(j)} C_{jv}^T \lambda_s + \sum_{s\in C_2(j)} \lambda_s,$$

where $C_1(j)$ and $C_2(j)$ are sets of cut joints related to body j.

Now the equation becomes

$$\sum_{j=1}^{n} \Phi_j^T F_j = 0,$$

and the Jacobian matrix can be derived as a particular case of Eq.(2).

$$J_{km}^q = S'_{\mathbf{F}k}\big|_{k=m} + S_k^{*T}\left(\tilde{\mathbf{F}}_{\pi}\big|_{m\in J(k)} + \tilde{\mathbf{F}}_r\big|_{m\in B(k)-\{k\}} + \mathbf{F}_{Rkm}\right)S_m^*, \quad \mathbf{F}_k = \sum_{j\in B(k)} F_j^* .$$

5. Approximated Jacobian matrices

Consider an implicit multistep method for numeric integration of Eq.(1) according to the following finite differences:

$$q_i = q_i^{\,p} + \delta q_i, \quad \dot{q}_i = \dot{q}_i^{\,p} + \delta q_i/\alpha, \; \ddot{q}_i = \ddot{q}_i^{\,p} + \delta q_i/\beta^2, \; i = 1,2\ldots, \qquad (3)$$

where the superscript ^{p} denotes predictions, δq_i is the unknown corrector, and the coefficients α,β (index i is omitted) are proportional to the step size h_i, e.g. for the Park method $\alpha = \beta = 0.6h_i$ [3].

Substituting Eq.(3) in Eq.(1) and linearization of the equation yield

$$\beta^2 f(\ddot{q}_i^{\,p}, \dot{q}_i^{\,p}, t) + J\delta q_i = 0$$

with the Jacobian matrix $J = M + J^v\beta^2/\alpha + J^q\beta^2$.

A useful simplification consists in neglecting the term J^q. Really, if the integration step is small and $\alpha << 1$, the inequality $\|J^v\|/\alpha >> \|J^q\|$ is valid very often. In this case the approximated matrix can be found as $J \approx M + J^v\beta^2/\alpha$. This formula can be useful, if the equations of motion are stiff due to dissipative forces.

Another important case concerns *slow* motions of stiff multibody systems, when the stiffness of equations of motion is caused by separate applied forces [3]. If the system motion is slow, the mass matrix is nearly constant, and inertia forces are small. For such cases the calculation of an approximated Jacobian matrix taking into account stiff components of applied forces proved to be very efficient. The simplified single block of the matrix is

$$J_{km} = M_{km} + S_k^{*T}(\mathbf{F}_{Vkm}\beta^2/\alpha + \mathbf{F}_{Rkm}\beta^2)S_m^* .$$

6. References

1. Andrzejewski, T, Bock, H. G., Eich, E. and von Schwerin, R. (1993) Recent advances in the numerical integration of multibody systems, in W. Schiehlen (ed.), *Advanced Multibody System Dynamics – Simulation and Software Tools*, Kluwer Academic Publishers, Dordrecht, pp. 127-151.
2. Featherstone, R. (1987) *Robot Dynamics Algorithms*, Kluwer Academic Publishers, Boston.
3. Pogorelov, D. (1998) Differential-algebraic equations in multibody system modeling, *Numeical. Algorithms* **19**, 183-194.

NONLINEAR CONTROL ALGORITHMS FOR MECHANISMS OF PARALLEL STRUCTURE

L.A. RYBAK
Moscow State Institute of Steel and Alloys
Leninsky Pr., 4
117936 Moscow
Russia

1. Introduction

Space technological implementation is parallel to the necessity of devices and technological systems vibration isolation under space condition. The vibration affects and in particular in low frequency and infra-low-frequency spectrum violates a some crystal growth processes and microbiology processes. There exist the problems of building a special vibration proofing systems of space technological platform.

2. Platform Stabilisation System

The task of stabilization of a technological platform from influence is considered in inertial system of coordinates. For control of stabilization the mechanisms of parallel structure are used which can be presented as rigid bar varying length on the information from sensors. These mechanisms and the platform are shown in fig. 1. All six modules have rotated connection with a platform and basement. Different methods can be used to change the length of each rigid bar. The task of the control system is to organize the operation of six mechanisms so that the position of the platform in inertial system of coordinates remains invariable.

One of the vibration isolation modules is shown on a diagram in fig. 2. The sensor of relative moving, accelerometer of the basement and platform are used in system. The signals from sensors act on a regulator. The control signal from a regulator is filtered and moves on an input of the executive engine. The electric motor through the transmission mechanism results a platform in a relative movement. For stabilization of speed of rotation of the engine the local feedback of an integrating type is used.

The task of stabilization of absolute coordinate of a platform cannot be decided as trivial by introduction of a feedback. In this case it is necessary to have the sensor of this coordinate, which in system is not present. Attempt to add such gauge is insoluble if to consider, for example, that the basement moves together with a platform with constant speed in some inertial system of coordinates. It is possible to put a task in another way: it is necessary to supply zero importance of acceleration on a platform.

Such task has infinite set of the decisions. It is possible to imagine two physical bodies, which move with different speeds without acceleration. Nevertheless, any of such decisions really cannot exist, it is necessary to take into account restrictions of a platform movement.

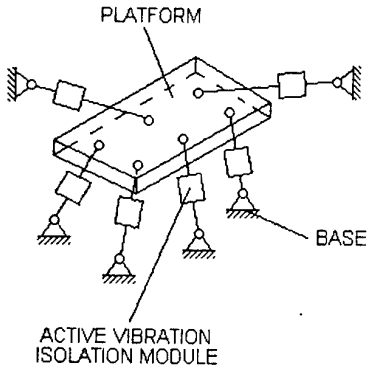


Fig. 1. Vibration isolation system for a platform in three-dimensional space

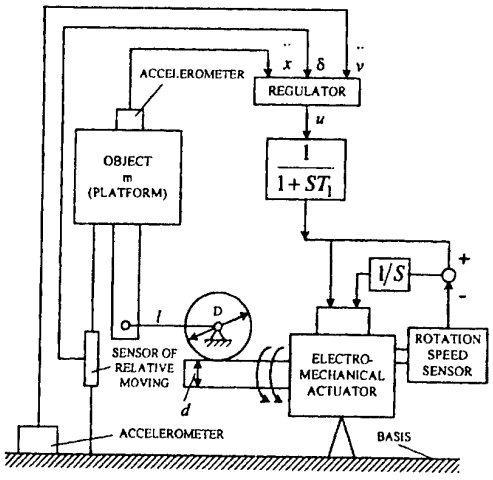


Fig 2. One-dimensional vibration isolation system.

Thus, we come to the following statement of a task: it is required to supply a minimum of acceleration on object with the limited moving.

To decide a task of synthesis of desirable system we shall proceed to the simplified model. The assumption is accepted, that all connections absolutely rigid, weight of a platform does not vary. With absence of a movement the system is tolerant to dynamic forces enclosed on object, only movement of the basis results a platform in a relative movement.

Let's solve a task of construction of digital system of stabilization. For a discrete regulator the control function is constant between the moments of switching. The executive mechanism is considered ideal. The speed of relative moving of a platform is proportional to a control signal. Thus, the function of acceleration at the moment of switching will have indefinitely large breaks. Hence, with transition to digital control it is necessary to filter a regulator signal. As the filter it is possible to take into account inertial properties of the executive mechanism, but it can be separate device.

Equation system for a continuous case of initial system has of the form of:

$$\begin{cases} \ddot{\delta} + \omega_1 \dot{\delta} = \omega_1 u \\ \ddot{x} = \ddot{\delta} + \ddot{\nu} \end{cases} \qquad (1)$$

x - absolute moving a platform,
δ - relative moving a platform,
ν - moving a basement,
u - control influence.

After performing a transition to state-space model and introducing variables $\theta(i) = \dot{\delta}(i)$, $x(i) = \ddot{x}(i)$, $\varphi(i) = \ddot{\nu}(i)$ the following system of the equations is received.

$$\begin{cases} \delta(i+1) = \delta(i) + a_{12}\theta(i) + b_1 u(i) \\ \theta(i+1) = +a_{22}\theta(i) + b_2 u(i) \\ x(i+1) = a_{32}\theta(i) + b_3 u(i) + \varphi(i+1) \end{cases} \quad (2)$$

где $a_{12} = \frac{1}{\omega_1 T}\left(1 - e^{-\omega_1 T}\right)$, $a_{22} = e^{-\omega_1 T}$, $a_{32} = -\omega_1 T e^{-\omega_1 T}$,

$b_1 = 1 - \frac{1}{\omega_1 T}\left(1 - e^{-\omega_1 T}\right)$, $b_2 = 1 - e^{-\omega_1 T}$, $b_3 = \omega_1 T e^{-\omega_1 T}$

3. Control System Synthesis

Call attention, that the received system (2) is system of the second order. The third equation is the equation of target value, acceleration on a platform. The function of acceleration $x(t)$ at the moment of switching control $t = iT$ has break limited on value. The third equation gives an estimation of acceleration $x(i)$ at moments of time, following behind switching of control. Therefore to use it for modeling a feedback on acceleration on a platform it is impossible. It is possible to take advantage of the following reception. Let program of control is $u(i) = kx(i)$. From system (1) we shall receive $u(i) = \frac{k}{1 - k\omega_1}\left(\varphi(i) - \omega_1\theta(i)\right)$. Similarly from some desirable feedback $u(i) = k\theta(i)$ we can proceed

$$u(i) = \frac{k}{\omega_1(1-k)}\left(\varphi(i) - x(i)\right) \quad (3)$$

Execute calculation of control, which will ensure invariance to acceleration of basis $x(i+1) = 0$. From last system equation (3) received

$$u(i) = -\frac{a_{32}\theta(i) + \varphi(i+1)}{b_3} \quad (4)$$

To get value of acceleration on the basis $\varphi(i+1)$ on the subsequent step it is possible only with modeling. In real system it is impossible. Therefore it is necessary

approximately to estimate value $\varphi^*(i+1) \approx \varphi(i+1)$. The following approximation was used

$$\varphi^*(i+1) = 3\,\varphi(i) + 3\,\varphi(i-1) + \varphi(i-2) \tag{5}$$

The system with the received law of control is unstable. For maintenance of stability we shall enter into the law of control small amendment $\Delta_1 \sim 0$

$$u(i) = -\frac{a_{32}(1-\Delta_1)\theta(i) + \varphi^*(i+1)}{b_3} \tag{6}$$

The received control function provides a low acceleration level on the basis, but poorly takes into account the requirement of restriction of relative displacement. To add into control program more strict requirements to the level of δ, we shall add the feedback on relative displacement. Let's search for factors of the following control function

$$u(i) = k_1\,\delta(i) + k_2\,\theta(i) + k_1\,\varphi(i) \tag{7}$$

Find factors k_1, k_2, k_3 proceeding from criterion of optimization

$$J = \rho\sum_{i=1}^{n} X^2(i) + \sum_{i=1}^{n} \delta^2(i) \to \min, \tag{8}$$

ρ - weight factor, provided that on system moves harmonic influence with frequency f:

$$\varphi(i) = \sin(2\,\pi f T i) \tag{9}$$

For search of factors the classical gradient algorithm was used.

Results of mathematical modeling are resulted which was calculate with meaning of parameters $\omega_1 = 13{,}8\text{c}^{-1}$, $T = 0{,}01$c; The numerical values of elements of system (2) are a_{12}=0,0093, a_{22}=0,871, a_{32}= –12,02, b_1=0,00066; b_2=0,13; b_3=2,02. Weight factor (8) was taken, equal $\rho = 10$, influence frequency $f = 0{,}5$ Hz (7). The received factors of control function k_1= –0,0038, k_2=0,0850, k_3=–0,0839.

The comparative analysis of the control (6) and (7) has shown, that the smaller acceleration level on a platform is provided with the control program (6), but the control program (7) provides smaller meaning of relative moving and smaller time on the established mode. Let's try to use advantages of both of these methods of control. Enter the following logic of switching. With small relative moving we shall use the control program (6). If the relative moving leaves from the given border $|\delta(i)| > \varepsilon$, that is switched control (7). As soon as the relative moving enters into the given corridor $|\delta(i)| < \varepsilon_2$, control (6) again is switched. It is obvious, that it is necessary to choose $\varepsilon_1 \geq \varepsilon_2$. Results of simulating for switching borders $\varepsilon_1 = 2$, $\varepsilon_2 = 0{,}02$, adjustment Δ_1=0,02 and

with parameters above mentioned, are shown in a fig. 3.

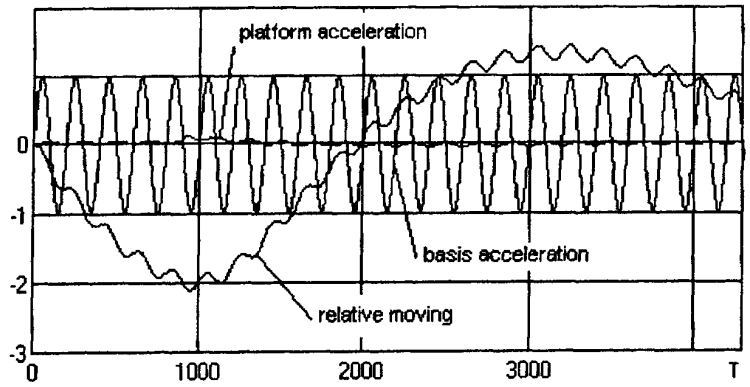


Fig 3. Time process for the nonlinear control program.

At an initial moment of condition zero, program of invariant control (6) is used. During the time, near 900 tacts, relative displacement gets to borders with the value -2 and is switching control (7). In this moment occurs a growing of platform acceleration, before this it was nearly is a zero. As soon as a relative displacement reaches borders with the value -1, at a moment of the time approximately 1600 tacts, is switched control (6).

Were they above considered two variants of building of regulator and their combination for the mode of nonlinear switching. Structures of transfer functions of control in these modes should be chosen based on the commmon sense considerations, but they may be not optimal. On the basis of analytical calculations [1,2,3] optimal control function was received for criteria (8)

$$u(i) = x_1\delta(i) + x_2\delta(i-1) + x_3\delta(i-2) + x_4\theta(i) + x_5\theta(i-1) + x_6\varphi(i) + \\ + x_7\varphi(i-1) + x_8\varphi(i-2) \qquad (10)$$

For searching factors x_m gradient algorithm was also used. Below are the values found for optimization criteria $J = \sum_{i=0}^{n} x^2(i) \to \min$:

x_1= –0,85, x_2=0,156, x_3=0,632, x_4= –4,30, x_5=5,19, x_6=0,661, x_7= –0,472, x_8= –0,326.

The results of modeling with the zero initial conditions are shown in a fig. 4.

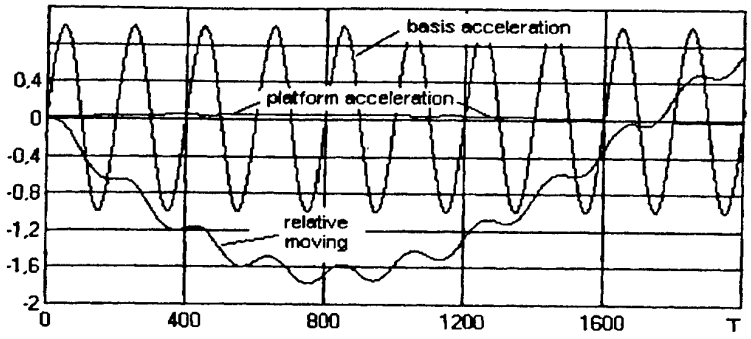


Fig 4. Time process under optimum transmission functions.

4. Summary

It is necessary to note that the character of behavior of system of stabilization for different control modes does not carry qualitative distinctions. In any the relative moving to the initial moment of time considerably leaves the considered variants from a reference value. With current of time it fades. To supply small relative moving with small acceleration on a platform essentially it is impossible. In any case the decision will have compromise character.

5. References

1. Rybak, L.A. and Siniov, A.V. (1994) Optimum regulator/controller synthesis of active vibration isolation system based on kinematics operation principle, *J. Machine Engineering* **6**, 23-30.
2. Rybak, L.A., Siniov, A.V. and Pashkov, A.I. (1998) *Synthesis of active vibration isolation systems on space objects*, Janus-K, Moscow.
3. Rybak, L.A. (2000) Active vibration-proofing systems of space technological platforms, *Proc. the VII International Congress on Sound and Vibration*, Garmisch-Partenkirchen, 367-370.

THE PROBLEM OF DYNAMIC CHAOS IN AUTOMATICALLY OPEN ON ORBIT OF LARGE-DIMENSION FOLDING REFLECTORS SPACE MIRROR ANTENNAS OF A TRUSS TYPE, EXECUTED AS OF SPATIAL MULTIBODY SYSTEMS

S. N. SAYAPIN
Mechanical Engineering Research Institute Russian Academy of Sciences
Moscow, Russia, 101830,

One of the main concepts of creation of large-dimension reflectors of space mirror antennas of a truss type executed as spatial beam link mechanisms with spring drives and elastic of a material reflecting radio wave, is built on their delivery in a folded position on the orbit and automatic deployable. The relevant advantage of the given concept as contrasted to by modular assembly on orbit through an astronaut or manipulator, is the capability of creation of space mirror antennas with the aperture from several tens up to several hundreds meters for one delivery by the launcher due to a high factor of their transformation (from 10 up to 50) [1].

In figure 1 the general view (a) and cyclogramme of deployment (b) on orbit of a large space mirror antenna is shown. Diameter of the uncovered mirror 1 makes 30 m, folded - 3 m.

At a phase *1* (the figure 1, b) is rotined a transit condition of a design of a space mirror antenna located under a fairing of a rocket 6. At a phase 2 the deployment of jacknife trusses of 2 focal unit 3 and their fixing implements. At a phase 3 the rise and fastening of bearings of 2 focal unit 3 on a jacknife mirror 1 in a transit condition is made. At a phase 4 opening-ups of a jacknife mirror 1 to automatic deployment, including deployment of a jacknife framework 5 and moving of a desktop with a jacknife mirror 1 in a transit condition on secure for deployment of a jacknife mirror of 1 spacing interval is made. At the phase of 5 automatic deployments of a jacknife mirror 1, including deployment of a jacknife framework of a truss and stretching on a working surface of a framework of a wireless of a reflecting grid, is made.

In a figure 2 the pieces of a jacknife truss framework with tetrahedral cells 1 in folded (a) and uncovered (b) positions are rotined. The framework contains jacknife rods 3, 4 and diagonal rods 5, which one paired by knotes 2.

At the same time, very great many of spring drives (more 6000), rods, socket joints (about 15000) and other configuration items of the antenna in combination to their irregular heating (cooling) and high difference of operation temperatures (from – 150^0 C up to $+150^0$ C), effect on pairs of friction of a high vacuum and strong spread of the characteristics of spring drives, is foregone results in formation of a strongly non-linear dynamic system and difficulty of maintenance of synchronization of deployment of configuration items. As a result of strong nonlinearity of properties and violation of

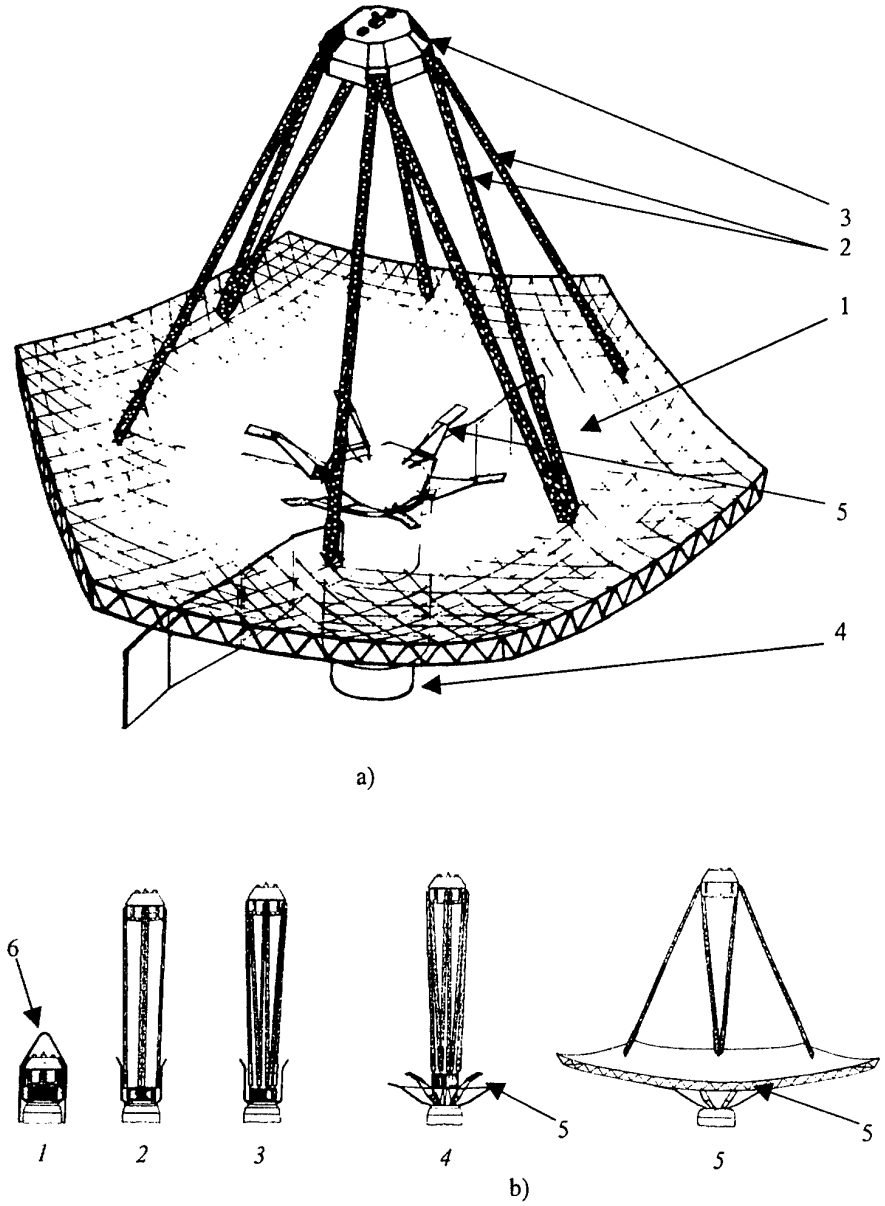


Figure 1.

motion stability of a system there is a transition to so-called dynamic chaos, the full unpredictability, contingency (i.e. stochastic process) motion of a system is characteristic practically for which one. A consequent of this lack is the low reliability and small probability of full deployment of a folding reflector of a mirror antenna in conditions of outside space. Thus, apparently, the degree of unpredictability of motion of a system directly depends on its sizes and, as a consequent, quantity kinematicly bound among themselves of mobile members.

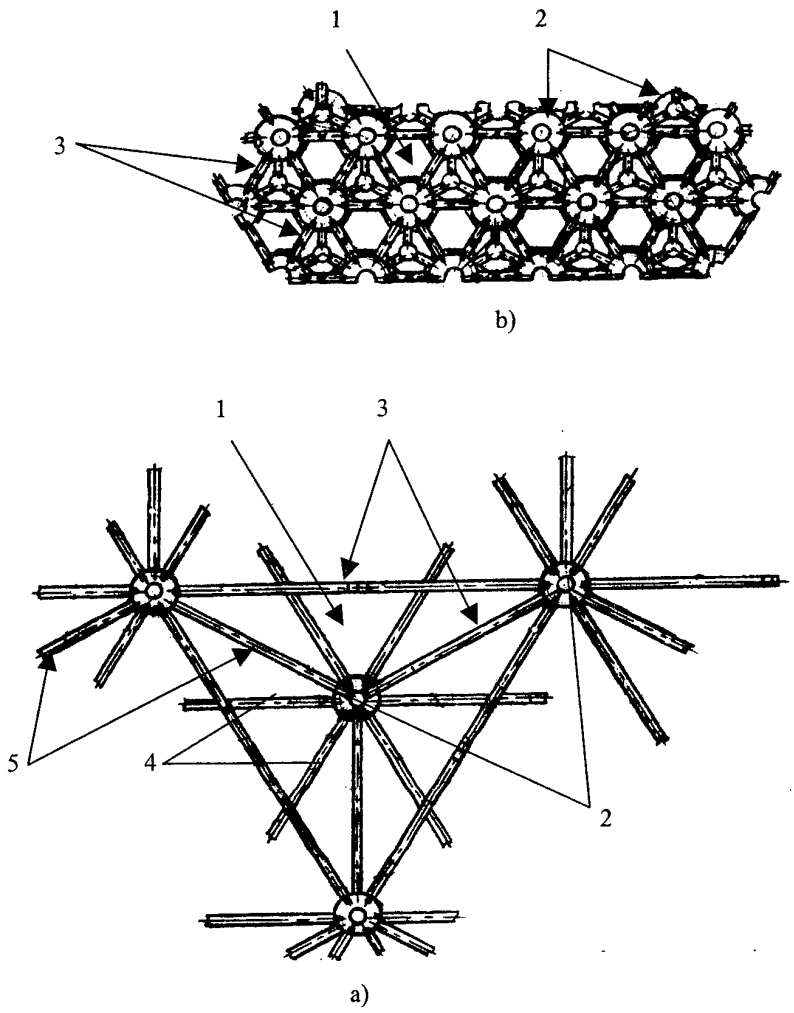


Figure 2.

The experiments have shown, that the steady automatic deployment of a jacknife truss framework can be reached at quantity of belts no more than 5, considering from center of a framework.

It is necessary also to mark, that the reliability of deployment of a jacknife framework of a mirror antenna depends on speed of its deployment. So, if his automatic deployment is made without constraining, the breaking of separate developing rods, and also their inexact deployment is watched as a result of a recoil and folding. And, to the contrary, in case of constrained deployment, the efforts developed in the final moment of deployment by spring drives, appeared poor for full deployment of all developing rods and tension on a framework of a wireless of a reflecting grid.

The introducing in a design of a folding reflector of a space mirror antenna of a truss type of members of controlled forced constrained deployment as kinematicly bound with it of a spatial pantograph [2], allows to supply reliable deployment of a system, but

simultaneously conducts to increase of its weight and consumption of power, that in a numbers of cases is unacceptable.

In figure 3 general views from above on a jacknife framework of a mirror of the large space antenna (a) and front elevation (view A) on his central part (б) by the way of spatial pantograph, is rotined. The jacknife framework of mirror antenna represents combined frame keeping kinematicly bound among themselves circumferential 1 and central 2 parts.

The deployment of a circumferential part 1 execute under operating of spring drives in jacknife rods, but central part 2 from the engine of deployment 3. Thus there is a constrained deployment of a circumferential part 1, and in case of possible jamming in her articulated joints - forced deployment, if necessary with reversing. It allows, to eliminate percussion actions of a circumferential part at her deployment, and to supply her controlled constrained and forced deployment with a capability of reversing if necessary. Thus the circumferential part provides to a framework demanded rigidity and lift capability, and central - constrained - forced deployment.

In the final moment of deployment of a jacknife framework it is required to supply a tension on his working surface of a wireless of a reflecting grid. For this purpose it is necessary to make automatic hooking up of padding spring drives. The problem was resolved as follows. Between each pair of knotes 4 (figure 3, b) working and non-working surfaces of a central part 2, arranged on one axis, the telescopic rods (central rods) with the stretched and captured spring (in figure 3, b, are not rotined), the butt ends which one are attached to inner sides of the conforming knotes 4. At deployment of a framework there is a rendezvous of inverse knotes 4 and accordingly decreasing of lengths of telescopic rods. In the final moment of deployment of a jacknife framework, when it is required to do stretching on his working surface of a wireless of a reflecting grid, there is an automatic actuating of devices of springs in ready condition, and the effort on deployment of a framework is sharply increased. In the uncovered position the mobile rods of telescopic central racks rise on horns, providing padding rigidity of a central part of the uncovered framework. In a figure 4 the full scale piece of a central part in processes of experimental improvement are submitted. In a figure 5 the full scale pieces of a circumferential part (a) of a jacknife framework of an mirror antenna and of a jacknife truss of bearings of the focal unit (b) in processes of experimental improvement are submitted. In figure 4 the telescopic rods 1 with the called above spring drives of a tension of a wireless of a reflecting grid are visible.

The reliability augmentation of deployment of a system can also be reached at the expense of exception of influencing of pull of a material reflecting radio wave of a material on process of deployment [2].

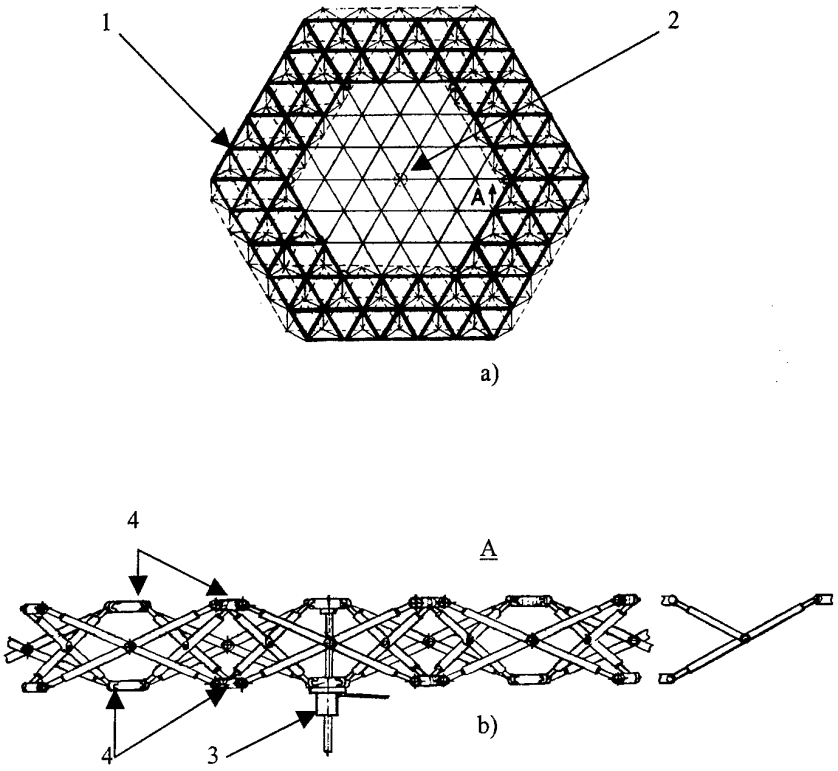


Figure 3.

However, the main source of unpredictability is the folding framework of a reflector of the antenna.

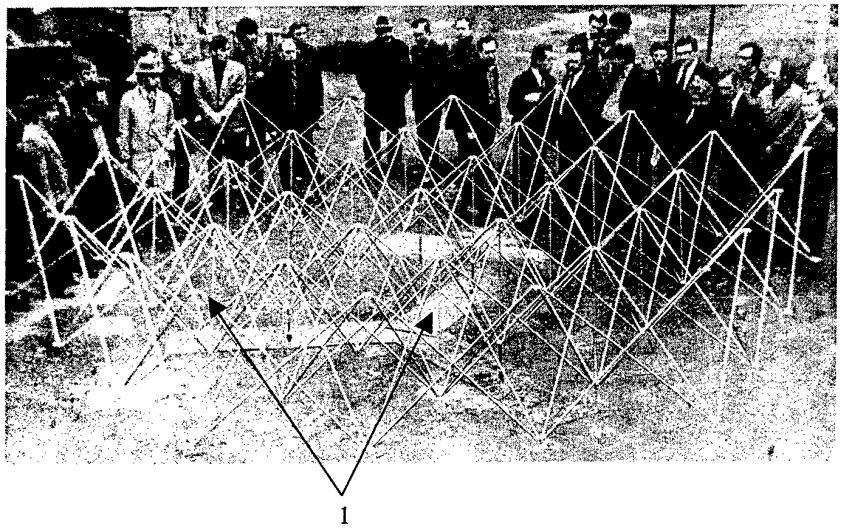


Figure 4.

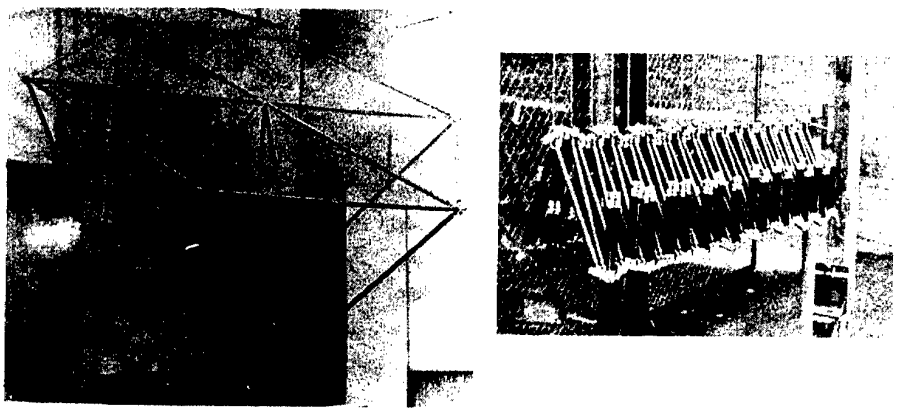


Figure 5.

Thus, the problem of dynamic chaos in automatically deployable on orbit large-dimension folding reflectors of space mirror antennas of a truss type is actual.
It is experimentally on breadboards, introduced in figure 4 and other, is rotined, that the imposing definitely of vibrational effect on a design of the folding mirror space antenna of a truss type reduces a level of unpredictability and improves reliability also stability of process her deployment [2].
By the writer it is offered to execute a system of vibrational effect on a bulky space mirror antenna of a truss type by the way of fissile transient truss of a space vehicle [3]. Thus, the transient truss run ins a view of a fissile gantry Stewart's [4] with six degrees of freedom. The conducted full scale experiments have shown good outcomes. The control of a gantry Stewart's in real-time mode is made from a neuronal computer [3, 5].

References

1. Gwamichava, A.S. (1981) Ampere-second. Problems of creation of folding space antennas. Transactionses CSRIDSC. Moscow, p.p. 29-36.
2. Sayapin S.N., Siniov A.V. The problem dynamics of chaos in automatic open of folding reflectors of space mirror antennas of truss type. XIII Symposium "Dynamics vibroimpact (strongly nonlinear)systems". Moscow-Zvenigorod 13-19 May 2001y. Thesis of reports. Moscow-Zvenigorod, 2001, p.p. 83-84.
3. Sayapin S.N., Siniov A.V., Trubnicov A.G. Way suppress hindrances from oscillations elastic of construction space deployable antennas in process exploitation and system for his realization. Patent of Russian Federation on invention No. 2161109, international class of inventions: B64G1/00, 1/22, 3/00, 1999. 18p.
4. Stewart D. A platform with six degrees of freedom//Proc. Inst. Eng. 1965-66. Vol. 180. Num. 15. Pt 1. Pt 1. P. 371 – 386.
5. Sayapin S.N., Siniov A.V., Galushkin A.I. "The system of active vibration protection and high-precision pointing of high-precision large deployable antennas space radio telescope with application neurocomputer control". Works VII All-Russian conference " The neurocomputers and them application NCA-2001" with international participation. Moscow 14-16 February 20001 y. ISBN 5-201-09606-9. Pp. 184-192.

DESIGN OF REDUNDANT PARALLEL ROBOTS BY MULTIDISCIPLINARY VIRTUAL MODELLING

Z. ŠIKA, M. VALÁŠEK, V. BAUMA, T. VAMPOLA
Department of Mechanics
Faculty of Mechanical Engineering
Czech Technical University in Prague
Karlovo nám. 13, 121 35 Praha 2, Czech Republic
E-Mail: valasek@fsik.cvut.cz , sika@fsik.cvut.cz

1. Introduction

The paper deals with the description of design methodology for redundant parallel robots based on multidisciplinary virtual modelling. The redundant parallel robots means redundantly actuated parallel robots. The parallel robots have many advantages as low moving masses, higher stiffness by truss structure, all drives on the frame, but they suffer by many problems like appearance of singularities and thus smaller workspace, collisions of links. These drawbacks of parallel structures can be removed by the principle of redundant actuation [4, 1]. This means that the platform is supported and driven by more bars with drives than the necessary number of DOFs. This principle not only deletes the singularities from workspace as more combinations of links in number of DOFs are not simultaneously in singular positions, but it brings further advantages, especially increased and more uniform dynamic capabilities, stiffness, accuracy.

The design of redundant parallel robots is an example of particularly complex design problem. The mutual dependencies of all parameters and components are especially large. The successful design methodology is possible only using virtual models and design complexity decomposition.

The used virtual models cover both mechanical including control and geometric properties. During the design there are conflicts between geometrical dimensions of robots and corresponding mechanical properties. The conflict includes collisions of robot links, non-existence of geometrical solutions of kinematics and insufficiency of mechanical properties like stiffness, dynamics, dexterity, accuracy etc. The design process has been resolved into three hierarchical levels. Each of these levels is characterized by certain problem simplification and special design conflict which should be resolved within the level.

2. Design Methodology

The design methodology of redundant parallel robots [1] follows the general engineering design methodology described in [2]. The design process is a hierarchical process as the technical products consist of hierarchy of components. The design process repeats the same outline at each design level. It consists of three nested loops:

- Selecting the lower level components from which the solution will be built.
- Proposing the structural arrangement of the selected components.
- Calculating parameter values so that the solution is complete, i.e. all requirements and constraints are fulfilled.

These three nested loops of component choice, structural arrangement and parameter choice also correspond to the nested design iterations and nested design optimisation. The component choice in the case of parallel robots means the decision about the fully parallel or hybrid concept, about the redundant/non-redundant concept, about the kind of link actuators, about the planar/spatial version of joints, about the kind of actuators (electrical/hydraulic, moving screw/direct electrical drive etc.), about the way of measurement etc. The structural design means the decision about the considered shape of components, about the way of their interconnections etc. After the structural design all decisions are transformed into numerical values of parameters. Their values are evaluated in terms of requirements and constraints.

The solution uses parts of mechatronic design methodology [3]. The most important methodological approach is the search for **Ideal Final Solution** and **Conflict Resolution** instead of conflict compromise that solves the given problem despite different conflicting constraints. This approach is looking for solutions that have advantageous values in criteria previously conflicting instead of just looking for tolerable compromise. The concept of redundant actuation is an example of such solution that keeps all advantages of parallel structures, removes problems with singularities and even improves the variations of main mechanical properties [4 - 6]. Certainly such principle is not found for each design task but in many cases just the idea of looking for ideal solution helps to overcome the local compromises. All steps of design and optimisation of robot properties has to be driven by concrete technological target of future machine from the very beginning state of design process. On the other hand "space of considered possibilities" should be held as wide as possible.

Specifically in case of redundant parallel robots the design process has been resolved into three hierarchical levels. Each of these levels is characterized by certain problem simplification and **special design conflict** which should be resolved within the level. (Quasi)optimum variants obtained as the best results of foregoing design optimisation level serve as starting variants for optimisation within the consecutive level. Certainly in the robot design there is mutual dependence between all parameters and thus feedback between levels is necessary. However mentioned decomposition into three subsequent design conflicts enables reasonably to simplify the design process:

- **Level of Geometric Conflicts:** Important properties of robot being designed besides the geometric requirements of DOFs, workspace and dexterity are represented by simple geometric conditions. For example requested limits of stiffness and modal properties are taken into account by some conditions for robot leg thickness, build-up spaces for real joints or robustness of machine frame. Optimisation of robot structure and dimensions try to harmonize several geometric requirements that are on the first try contradictory:
    1. Workspace without collisions and kinematic singularities should be maximized.
    2. Ratio between total build-up space of machine and useful (technological) workspace should be minimized.
    3. Dimensions or build-up spaces of important machine elements should be sufficient.
    4. Dexterity should be optimised (maximization and uniformity in workspace).
- **Level of Structural Conflicts:** The structural conflict comprehends more precisely formulated conflict between structural (stiffness and modal) properties of the whole machine and accessible dynamics (velocity, acceleration, jerk) of robot end-effector. Mutual interrelations of these properties are very complex and in addition other important aims of machine designers (like accuracy for higher speeds of operations) are heavily influenced by them. Basic requirements are as follows:
    1. Accessible dynamics (velocity, acceleration, jerk) of robot end-effector should be maximal and uniform for representative trajectories within the workspace.
    2. The first eigenfrequencies of the robot should be as high as possible and uniform for all possible robot positions in the workspace.
    3. Cumulative stiffness measured on the end-effector should be maximal and uniform for all possible robot positions in the workspace.
- **Level of Actuation Conflicts:** Behaviour of the whole machine depends on dynamic interactions among mechanical parts, electrical or hydraulic actuators and feedback control loops of actuators. Simulation of complex mechatronic system must be performed in order to predict potential problems arising here. Thoroughgoing fulfilment of previous two design levels is crucial for efficiency of final complex tuning. Basic requirements are as follows:
    1. Control loops must be stable without troublesome vibrations.
    2. Control loops of actuators must be tuned in order to make drives as dynamic as possible. Technological times of production should be minimized.
    3. Energy consumption of drives necessary for production should be minimized.
    4. Accuracy for high speed operations should be maximized.

The applied design methodology is heavily based on the efficient computational tools for mapping robot design parameters into design criteria (requirements and constraints) and following **multiobjective optimization** of the robot parameters like dimensions, drive parameters, control parameters. For mechanical properties there have been developed computational tools based on global dynamics [7]. There are also very important visualization tools especially for multiobjective design. For the design of redundant parallel robots the following computational tools are used:

1. **Workspace, dexterity and collisions evaluation.** The crucial property of the robot is the geometric and kinematic synthesis. The size of workspace limited by geometric and collision constraints are evaluated and mapped in each position. The efficient analysis of collisions of arbitrary bodies has been implemented. The basic entities for the collision detection are general cuboids. The complex bodies are replaced-approximated by the composite bodies composed from many cuboids. The problem of collisions of cuboids has been solved in two stages. The first fast step evaluates potential possibility of collision. The second stage is initialised whenever the collision cannot be excluded. The penetrations of edges of one body and surfaces of second body have been detected during the second detailed stage of analysis. Collision can be visualized in 3D or 2D in basic planes of coordinate system. Besides that the occurence of singularity positions and generally the manipulability of the robot are evaluated.
2. **Stiffness and eigenfrequency (modal) evaluation.** The accuracy is dependent on the robot stiffness. There are evaluated the maps of robot stiffness and eigenfrequencies.
3. **Dynamic capability evaluation.** The limitations of dynamic capabilities of drives are transformed into the areas of accessible accelerations and velocities at the points of selected trajectory using methods of global dynamics [7]. Choosing several trajectories like straight lines with different slopes across the workspace and the circles with different radius enables to map the overall dynamic capabilities of the robot.
4. **Force transmission evaluation.** The accessible accelerations and velocities from previous step are achieved through particular driving forces. Their determination due to the actuator redundancy is not straightforward and simple [8]. The driving forces and corresponding reaction forces in joints and structural elements are transmitted through the robot structure and this force transmission and distribution is important for dimensioning of robot structural elements.
5. **Kinematic and elastostatic accuracy evaluation.** The accuracy is essential robot property. It is influenced by the properties of encoders and by the robot stiffness in relation to the external applied forces.
6. **Control design.** The control design is done by the methodology design by simulation.
7. **Overall simulation.** The designed properties are verified within overall simulation where especially the multibody, elastic and control properties are investigated in deep interaction.
8. **Multiobjective optimization.** The above listed performance criteria as well as others are subjected to the multiobjective optimization using the design parameters of the robot. The Pareto sets of conflicting criteria are computed and visualized.

3. Design Case Study

The case study is devoted to the investigation of improvement of mechanical properties of Sliding Delta robot (*Fig. 1*), also called Uran. The robot Octaslide (*Fig. 3*), the more complex (6 DOF motion of end-effector) modification of original robot has been designed as well. The main potential of improvement is based on the application of principle of redundant actuation. It brings for Sliding Delta mainly improvement of stiffness and dynamics, for Octaslide especially the elimination of singularities. The design was performed within many iteration loops. It is difficult to reconstruct the content of all of them in details. However all three nested levels of design conflicts were investigated and solved as follows. There were used computational tools mentioned above.

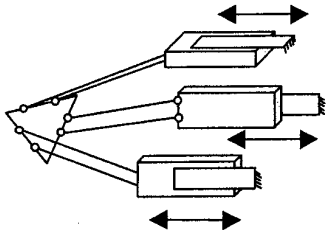
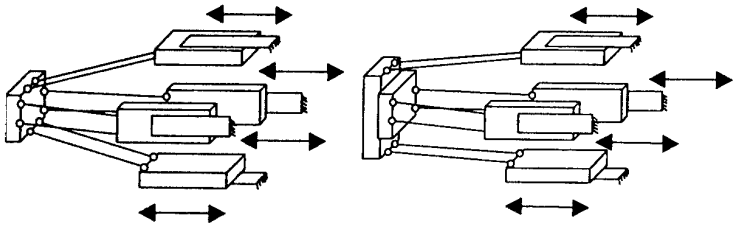


Figure 1. Original Sliding Delta (Uran) robot with sliding joints

3.1. GEOMETRIC CONFLICTS AND THEIR SOLUTION

Initially the original structure from *Figure 1* had been extended into the redundant version on *Figure 2a*. Then the structural properties of designed robot were represented by simple geometric conditions. The critical value was the diameter of the legs in order to achieve reasonable stiffness.

Simultaneously the lengths of legs had to be kept in values comparable with non-redundant version again due to comparable stiffness. The critical issue was the computation of accessible workspace due to the collisions and improvement of dexterity. The problems of finding parameters (dimensions of platform and legs) for simultaneously good workspace and dexterity could not been resolved on the level of parameter values and have lead finally to the modification of the structure from initial version (*Fig. 2a*) into the final one (*Fig. 2b*).



(a) (b)

Figure 2. The redundant Sliding Delta: initial (a) and final (b) concepts

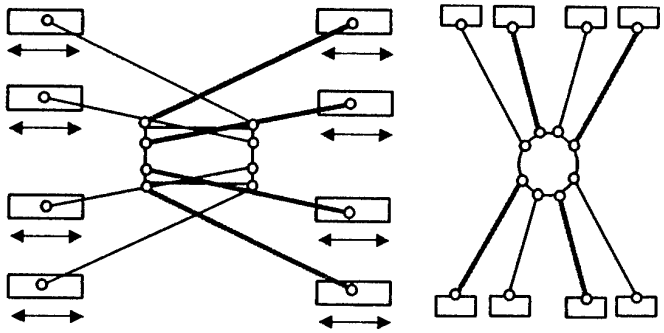


Figure 3. Scheme of robot Octaslide

3.2. STRUCTURAL CONFLICTS AND THEIR SOLUTION

The structural design is about the resolving of conflict between stiffness and dynamics. First the stiffness of both non-redundant and redundant parallel structures are evaluated (*Fig. 4*). It means the stiffness in both directions in the plane of motion and in the direction of sliding drives. This clearly demonstrates the significant improvement of stiffness by almost 50%.

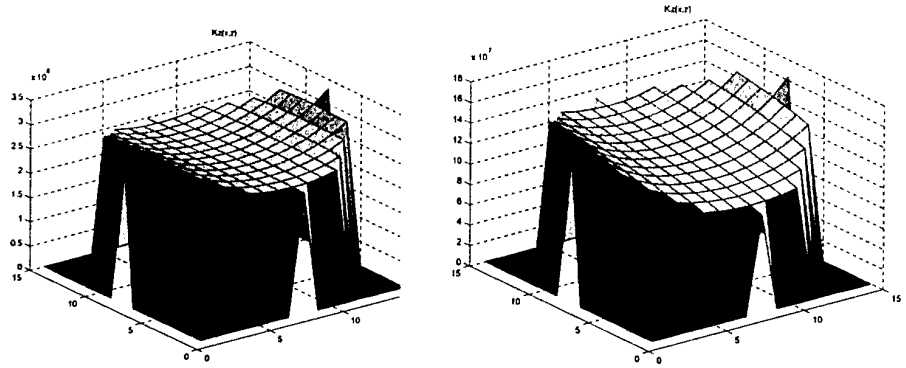


Figure 4. The comparison between stiffness of non-redundant (right) and redundant (left) parallel structure for one planar section in the workspace

Second there is investigated the dynamic capabilities. The limitations of dynamic capabilities of drives

are transformed into the areas of accessible accelerations and velocities at the points of selected trajectory. Choosing circular trajectories with different radius the dynamic capabilities were evaluated at the radius 0.3 m and at the radius 0.6 m (*Fig. 5*). There are plotted the accessible accelerations versus accessible velocities on the circular trajectories for both non-redundant and redundant versions. Again the redundant actuation has proved significant improvement of dynamics by about 20%.

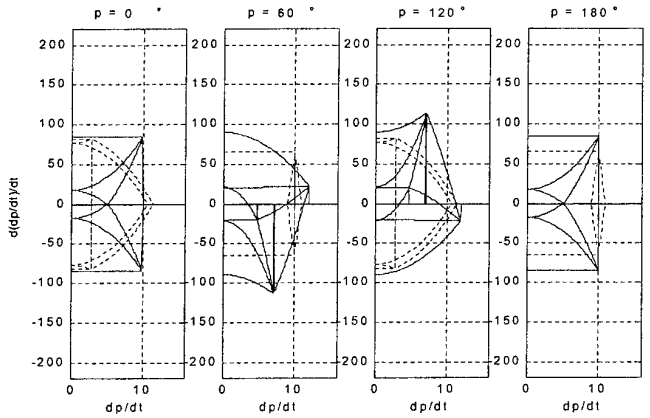


Figure 5. The dynamic capabilities of non-redundant (dash lines) and redundant (full line) parallel structures during the circular motion with radius 0.6 m (accessible acceleration on velocity in 4 positions)

Then the conflict between stiffness (eigenfrequencies) and dynamics of end-effector (mass) for different variants of dimensions has been solved by multiobjective parameter optimisation. In short on the border of space of possible solutions (Pareto set) increasing stiffness means increasing mass and decreasing acceleration capabilities. The genetic algorithms have been used for this task. Each point (*Fig. 6*) represents one variant of setting of robot dimensions.

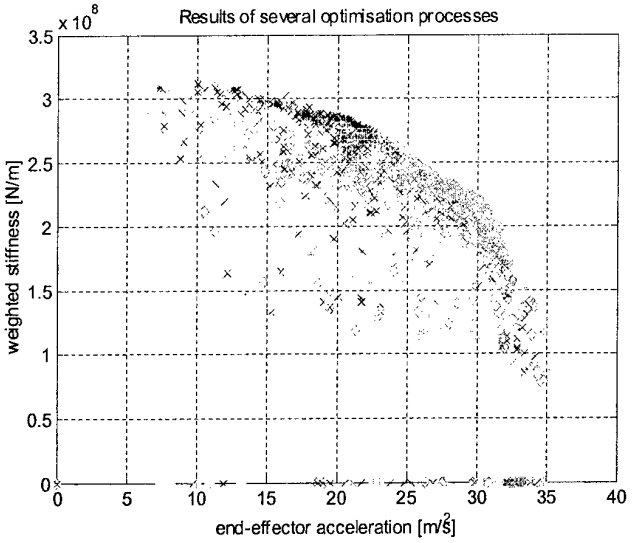


Figure 6. Results of multiobjective parameter optimisation of stiffness and dynamics (results of several optimisation processes displayed together)

3.3. ACTUATION CONFLICTS AND THEIR SOLUTION

The drive concept must be completed from the point of view of required dynamics, kinematics, dynamic accuracy and control strategy. The simplified scheme (*Fig. 7*) describes two nested loops for slider position and motor angular velocity feedbacks. The end-effector position measurement can be considered for the upper-most feedback loop, nevertheless its practical realisation is not easy. The tuning of control gains completes the utilization of previously designed mechanical properties. The overall

simulation is the final test of the whole redundant parallel robot conceptual design.

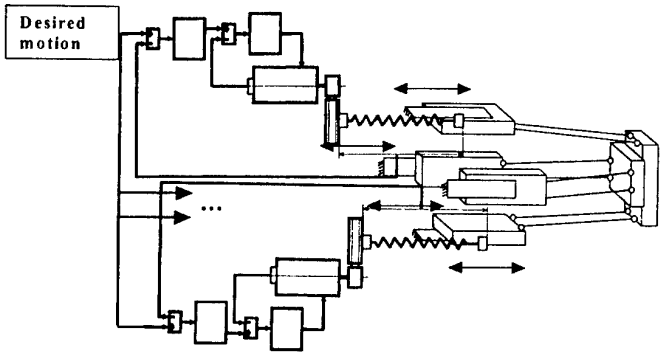


Figure 7. Simplified scheme of complex dynamic model including feedback control loops

4. Conclusions

The paper has briefly investigated the role of multidisciplinary virtual modelling for efficient design of complex mechatronic machines. It has been demonstrated on the design methodology for redundant parallel robots. The virtual models cover both mechanical and geometric properties. They are based on multibody models despite they cover different properties. The basis for these virtual models is the decomposition of design process. The design process has been resolved into three hierarchical levels. Each of these levels is characterized by certain problem simplification and special design conflict which should be resolved within the level. Specific virtual models are necessary for each level. The computational tools related to these virtual models enable to parameterise the main design conflicts and solve them using multi-objective parameter optimisation. Proposed hierarchical methodology based on multidisciplinary virtual modelling proved to be useful and efficient for the design of complex mechatronic machines.

The proposed design methodology based on multidisciplinary virtual modelling has been demonstrated on the design case study of redundant Sliding Delta and Octaslide robots. They also demonstrate the application of the principle of redundant actuation that leads to the development of new robot parallel structures with promising properties.

Acknowledgment. The authors appreciate the kind support by the grant J04/98:212200003 "Development of algorithms of computational mechanics and their application in engineering".

5. References

1. Valášek, M., Šika, Z., Bauma, V., Vampola, T. (2001) Design Methodology for Redundant Parallel Robots, in *Proc. of AED 2001, 2nd Int. Conf. on Advanced Engineering Design*, Glasgow, 243-248.
2. Zdráhal, Z., Valášek, M. (1996) Modeling Tasks in Engineering Design, *Cybernetics and Systems* **27**, 105-118.
3. Valášek, M. (1998) Mechatronic System Design Methodology - Initial Principles Based on Case Studies, in J. Adolfsson and J. Karlsen (eds.), *Mechatronics 98*, Pergamon, Amsterdam, 501-506.
4. Šika, Z., Valášek, M., Miláček, S., Bastl, P. (1997) Synthesis and Analysis of Planar Redundant Parallel Robot, in J. Angeles and E. Zakhariev (eds.), *Proc. of NATO ASI Computational Methods in Mechanisms*, Varna, Vol. II, 353-362.
5. Kim J., Park F.C., Ryu S.J., Kim J., Hwang J.C., Park C., Iurascu C.C. (2001) Design and analysis of a redundantly actuated parallel mechanism for rapid machining, *IEEE Trans. On Robotics and Automation*, Vol. 17 (4), 423-434.
6. Kock, S., Schumacher, W. (1998) Regelungsstrategien für Parallelroboter mit redundanten Antrieben, in *VDI-Fachtagung "Neue Maschinenkonzepte mit parallelen Strukturen für Handhabung und Produktion", VDI Berichte Nr. 1427*, Braunschweig, 155-164.
7. Valášek, M., Šika, Z. (2001) Evaluation of Dynamic Capabilities of Machines and Robots, *Multibody System Dynamics* **5**, 183-202.
8. Valášek, M., Šika, Z. (2001) Analysis of Required Driving Force Distribution in Redundant Parallel Robots, in *Proc. of AED 2001, 2nd Int. Conf. on Advanced Engineering Design*, Glasgow, 258-263.

The NATO Science Programme

NATO Advanced Study Institute on

Virtual Nonlinear Multibody Systems

Prague, Czech Republic

June 23-July 3, 2002

University of Stuttgart
Institute B of Mechanics

Czech Technical University in Prague
Department of Mechanics

http://mechatronic.fsik.cvut.cz/ASI_NATO

NATO Advanced Study Institute (ASI) on Virtual Nonlinear Multibody Systems Prague June 23-July 3, 2002

Place: Hotel Krystal, Czech Technical University in Prague, Prague, Czech Republic

Organizing commitee:
Chairman: Prof. W. Schiehlen, University of Stuttgart, Stuttgart, Germany
Co-chairman: Prof. M. Valášek, Czech Technical University in Prague, Prague, Czech Republic

Members:
Prof. A. Shabana University of Illinois at Chicago, Illinois, USA
Prof. E. Kreuzer Technical University Hamburg-Harburg, Hamburg, Germany
Prof. J. Ambrosio IDMEC/IST, Instituto Superior Tecnico, Lisboa, Portugal
Prof. J. McPhee University of Waterloo, Ontario, Canada

Deadlines:
Paper abstract submission: January 31, 2002
Application for financial support: January 31, 2002
Paper acceptance: February 28, 2002
Financial support decision: February 28, 2002
Application for participation only: April 30, 2002
Full paper submission: May 15, 2002
NATO ASI: June 23 - July 3, 2002

Web: http://mechatronic.fsik.cvut.cz/ASI_NATO

Advanced Study Institute description
While not yet in the dictionary, the term "virtual reality" has taken on the meaning of the "computer generated perception of reality on the part of an involved human. The term "virtual reality" motivated the emerging use of the term "virtual prototype", suggesting both computer and human involvement in the process of design and testing of the prototype that do not exist as real object. The complex multibody systems, that are composed of rigid and flexible bodies achieving spatial motion and various complex tasks, are up-to date objects of the virtual prototyping. The real simulation of the multibody system action takes into account various phenomena and particular properties. The classical dynamics do not regard various nonlinear effects that appear as a result of the action of multibody systems, as well as their mutual interaction. The virtual prototyping and dynamic modeling of such systems are, from economical point of view, perspective realms of scientific investigations having in mind the huge expenses for their design and manufacturing.

Future research fields in multibody dynamics are identified as standardisation of data, coupling with CAD systems, parameter identification, real-time animation, contact and impact problems, extension to electronic and mechatronic systems, optimal system design, strength analysis and interaction with fluids.

Objectives of the Advanced Study Institute:
- To meet leading scientists in the field of nonlinear multibody system dynamics
- To create conditions for exchange of ideas and development of collaborative applied projects
- To give possibilities for perspective young researchers to present their achievements and to establish professional contacts
- Emphasise will be given to analysis, synthesis and stability of large flexible systems and problems related to vehicles, aircraft, spaceships, manipulators, robots and constructions imposed on action of earthquakes.

This long course (10 days) is intended to provide a comprehensive coverage of the whole subject in its multidisciplinary aspects. The course is intended for PhD holders and advanced PhD students as well.
Some background is assumed in applied dynamics and control.

The possible presentation of contributed papers of the participants is assumed.

Topics:

- Nonlinear dynamics models
- Unilateral constraints and contact
- Stability analysis and chaotic motions
- Data processing and visualization
- Control and optimization

Invited Lectures:

W. Schiehlen: Contact Problems And Wave Propagation In Nonlinear Multibody Systems
A. Shabana: Nonlinear Dynamics Of Multibody Systems With Generalized And Non-Generalized Coordinates
F.L. Chernousko: Snake-like Locomotions of Multilink Systems
J. Ambrosio: Impact of Rigid and Flexible Multibody Systems: Implications on the deformation description and contact models
M. Pascal: Numerical Simulation of Flexible Multibody Systems By Using A Virtual Rigid Body Model
P. Nikravesh: Flexible Body Model Reduction Techniques In Multibody Dynamics
F. Pfeiffer: Unilateral Multibody Dynamics
V. Berbyuk: Control And Optimization of The Semi-Passively Actuated Multibody Systems
M. Pereira: Optimization of Rigid-Flexible Multibody Systems With Application to Vehicle Dynamics and Crashworthiness
M. Lankarani: Analysis of External and Internal Impacts in Multibody Systems
E. Kreuzer: Multibody System Dynamics in Ocean Engineering
J. McPhee: Graph-Theoretic Modelling of Multibody Systems
E.J. Haug: Vehicle Virtual Proving Grounds
M. Valasek: Nonlinear Control Synthesis For Nonlinear Multibody Systems
W. Blajer: Geometrical Interpretation of Multibody Dynamics: Theory and Implementations
D. Bestle: Kinematic and Kinetic Analysis of Dynamic Systems, Optimization of Passive and Active Dynamic Systems
E. Zahariev: Modeling of Mutibody System Contact Dynamics
P. Eberhard: Contact Formulations for Finite Elements and Multibody Systems

Registration:

If you want to participate at the NATO ASI you must register at the NATO ASI secretariat. If you want to present a paper you must submit an abstract at the NATO ASI secretariat. If you need a financial support, you must fill in an application form and send it to the NATO ASI secreatriat. Please take into account deadlines of the NATO ASI.
It is also possible to register on-line.

Grants:

Special grants are available for students from NATO and Partner countries. Travel, accomodation and subsistence will be supported. Applications should be sent conference secretariat before January 31, 2002.

Accommodation:

Hotel, where accomodation and full board for participants is arranged, is located near the path from the Prague's airport into the city center. It is also close to the Conference center where the NATO ASI will be hold. There are both single rooms (30 Euro/day) and double rooms (45 Euro/day) with full board available.

The participants of NATO ASI should book accomodation at the NATO ASI secretariat.

Contact Information:

NATO ASI secretariat can be reached by following means:

Telephone: +420-2-24357491
Fax: +420-2-24916709
E-mail: natoasi@felber.fsik.cvut.cz
Web: http://mechatronic.fsik.cvut.cz/ASI_NATO

Address: Prof. Michael Valasek, NATO ASI, Dept. of Mechanics, Karlovo nam. 13, Praha 2, 121 35, Czech Republic

NATO ASI on
Virtual Nonlinear Multibody Systems
Prague June 23-July 3, 2002

Family name:

First name: Title:

Organisation:

Address:

Country:

Phone:

Fax:

E-mail:

I intend to submit a paper YES □ NO □
Tentative title of the paper:

Please return to the NATO ASI secretariat:

Address: Prof. Michael Valasek, NATO ASI, Dept. of Mechanics, Karlovo nam. 13, Praha 2, 121 35, Czech Republic

Fax: +420-2-24916709
E-mail: natoasi@felber.fsik.cvut.cz

NATO Advanced Study Institute on

Virtual Nonlinear Multibody Systems

Prague, Czech Republic, 23 June - 3 July 2002

Scientific Abstract

Multibody system dynamics is based on classical mechanics and its engineering applications ranging from mechanisms, gyroscopes, satellites and robots to biomechanics and vehicle engineering. Multibody systems dynamics is characterized by algorithms or formalisms, respectively, ready for computer implementation. The simulation of multibody systems demands for adequate dynamic models and takes into account various phenomena. Classical dynamics does not regard all nonlinear effects that appear as a result of the action of multibody systems, as well as their mutual interaction. The virtual prototyping and dynamic modeling of such systems are, from an economical point of view, perspective fields of scientific investigations having in mind the huge expenses for their design and manufacturing. Complex multibody systems composed of rigid and flexible bodies performing spatial motion and various complex tasks are up-to-date objects of virtual prototyping. As a result simulation and animation featuring virtual reality are most important. Recent research fields in multibody dynamics include standardization of data, coupling with CAD systems, parameter identification, real-time animation, contact and impact problems, extension to electronic and mechatronic systems, optimal system design, strength analysis and interaction with fluids. Further, there is a strong interest on multibody systems in analytical and numerical mathematics resulting in reduction methods for the rigorous treatment of simple models and special integration codes for Ordinary Differential Equation (ODE) and Differential Algebraic Equation (DAE) representations supporting the numerical efficiency. New software engineering tools with modular approaches improve the efficiency still required for the more demanding needs in biomechanics, robotics and vehicle dynamics. The scientific research in multibody system dynamics is devoted to improvements in modeling considering nonholonomic constraints flexibility, friction, contact, impact and control. New methods evolved with respect to simulation by recursive formalism, to closed kinematic loops, reaction forces and torques, and pre- and post-processing by data models, CAD coupling, signal analysis, animation and strength evaluation. Multibody system dynamics is applied to a broad variety of engineering problems from aerospace to civil engineering, from vehicle design to micromechanical analysis, from robotics to biomechanics. In particular, multibody dynamics is considered as the basis of mechatronics, e.g. controlled mechanical systems. These challenging applications are subject to fundamental research topics which were presented at the NATO ASI on Virtual Nonlinear Multibody Systems.

1. Datamodels

Within the multibody system community many computer codes have been developed, however, they differ widely in terms of model description, choice of basic principles of mechanics and topological structure so that a uniform description of models does not exist. The data exchange permits the alternate use of validated multi-body system models with different simulation systems.

2. Parameter identification

The parameter identification is an essential part of multibody dynamics. The equations of motion of mechanical systems undergoing large displacements are highly nonlinear, however, they remain linear with respect to the system parameters.

3. Optimal design

Due to development of faster computing facilities the multibody system approach is changing from a purely analyzing method to a more synthesizing tool. Optimization methods are applied to optimize multibody systems with respect to their dynamic behaviour.

4. Dynamic strength analysis

The results obtained in research on strength analysis of material bodies can be applied and combined with the multibody system approach.

5. Contact and impact problems

Rigid and/or flexible bodies moving in space are subject to collisions what mechanically means impact and contact. Contact problems usually include friction phenomena which are modelled by Coulomb's law.

6. Extension to control and mechatronics

The applied forces and torques acting on multibody systems may be subject to control. Then, the multibody system is considered as the plant for which the controller has to be designed. Today, mechatronics is understood as an interdisciplinary approach to controlled mechanical systems usually modelled as multibody systems.

7. Nonholonomic systems

The nonholonomic systems are of engineering interest in vehicle dynamics and mobile robots.

8. Integration codes

The dynamic equations of motion are presented as ODE or DAE. Efficient algorithms for numerical integration of these equations are of major importance.

9. Real time simulation and animation

Efficient and fast simulation is always desirable in computational dynamics but it is really necessary for hardware-in-the loop and operator-in-the-loop applications. There are two approaches to achieve real time simulation: high speed hardware and efficient software. Multibody system dynamics contributes to the efficiency of the software by recursive and/or symbolic formalism and fast integration codes.

10. Challenging applications

Multibody system dynamics has a broad variety of applications. In biomechanics the walking motion is an important topic. However, there are much more problem in biomechanics which can be modeled and solved by multibody dynamics. The applications are ranging from vehicle occupants to sport sciences. Multibody dynamics is also a solid basis for nonlinear dynamics. In particular, impact and friction induced vibrations show chaotic behaviour. The control aspects in multibody dynamics are getting more and more important. Vehicle, aircraft and spaceship dynamics and reliability have always been challenging applications. With respect to transportation systems a challenging application of multibody dynamics is the structural and occupant crashworthiness.

Programme of NATO Advanced Study Institute on Virtual Nonlinear Multibody Systems

Sunday , 23 June 2002

14:00 - 16.30 Registration
16.30 - 17.30 Opening Session

Monday , 24 June 2002

08:30 - 10:30 Lecture 1
W. Schiehlen
Virtual Assembly in Multibody Dynamics

11:00 - 13:00 Lecture 2
A. A. Shabana
Non-Linear Dynamics of Multibody Systems with Generalized and Non-Generalized Coordinates

Session 1 - Chairperson: E. Zahariev

14:30 - 14:50 J.-C. Piedbœuf, J. Kövecses, B. Moor, R. L'Archevêque:
Symofros: A Virtual Environment for Modeling, Simulation and Real-Time Implementation of Multibody System Dynamics and Control

14:50 - 15:10 I. Stroe
Multibody Systems with Holonomic and Non-Holonomic

15:10 - 15:30 A. L. Schwab, J. P. Meijaard
How to Linearize Your Non-Holonomic Multibody System

15:30 - 15:50 D. Bernier, A. Dequidt, E. Valdes
Symbolic and Systematic Multibody Modelling for Mechatronic Design

15:50 - 16:10 D. Talaba
A Particle Model for Mechanical Systems Simulation: A Model Based Overview of Multibody Systems Formulations

Session 2 - Chairperson: J. Mc Phee

16:30 - 16:50 D. Yu. Pogorelov
On Calculation of Jacobian Matrices in Simulation of Multibody Systems

16:50 - 17:10 G. V. Boiadjiev, D. B. Vassileva
Dynamic Sensibility of Systems with Redundancy

17:10 - 17:30 P. Steinbauer, M. Valášek, Z. Zdráhal, P. Mulholland, Z. Šika
Knowledge Support of Virtual Modelling and Simulation

Tuesday , 25 June 2002

08:30 - 10:30 Lecture 3
F. L. Chernousko
Snake-like Locomotions of Multilink Systems

11:00 - 13:00 Lecture 4
J. A. C. Ambrósio
Impact of Rigid and Flexible Multibody Systems: Deformation Description and Contact Models

Session 3 - Chairperson: J. A. C. Ambrósio

14:30 - 14:50 L. Hyncík
Multi-Body Human Model

14:50 - 15:10 M. P. T. Silva, J. A. C. Ambrósio
Solution of Redundant Muscle Forces in Human Locomotion with Multibody Dynamics and Optimization Tools

15:10 - 15:30 M. Wojtyra
Multibody Model of Human Walking

15:30 - 15:50 M. C. Tofan, S. I. Vlase, I. I. Micu
Parametric Identification of the Elastic Pole-Vaulting Pole

15:50 - 16:10 A. Eiber, H.-G. Freitag, C. Breuninger
Virtual Reconstruction of Impaired Human Hearing

Session 4 - Chairperson: M. Pascal

16:30 - 16:50 Z. Šika, M. Valášek, V. Bauma, T. Vampola
Design of Redundant Parallel Robots by Multidisciplinary Virtual Modelling

16:50 - 17:10 K. Beneš
A First Step towards FE Modelling of Ergonomics and Comfort

17:10 - 17:30 T. Yu. Figurina
Quasi-Static Motion of the Two-Link and Three-Link Mechanisms along a Horizontal Plane

Wednesday , 26 June 2002

08:30 - 10:30 Lecture 5
M. Pascal
Numerical Simulation of Flexible Multibody Systems by Using a Virtual Rigid Body Model

11:00 - 13:00 Lecture 6
P. E. Nikravesh
Model Reduction Techniques in Flexible Multibody Dynamics

Session 5 - Chairperson: M. S. Pereira

14:30 - 14:50 A. Mikkola, A. Shabana
Development of Plate and Shell Elements for Flexible Multibody Applications

14:50 - 15:10 O. N. Dmitrotchenko
Efficient Simulation of Rigid-Flexible Multibody Dynamics: Some Implementations and Results

15:10 - 15:30 N. Kobayashi, M. Watanabe, T. Irie, K. Kozono
Dynamics and Stability of Spaghetti and Reverse Spaghetti Problems Coupled with Fluid Force

15:30 - 15:50 L. Kübler, P. Eberhard
A Formulation for Flexible Multibody Systems with Mixed Cartesian and Relative Coordinates

15:50 - 16:10 S. I. Vlase, M. C. Tofan, I. A. Goia
Some Aspect of Finite Element Analysis of Flexible Multibody Systems

Session 6 - Chairperson: F. Chernousko

16:30 - 16:50 J. Valverde, J. L. Escalona, J. Mayo, J. Domínguez
Dynamic Analysis of a Light Structure in Space: Short Electrodynamic Tether

16:50 - 17:10 S. N. Sayapin
The Problem of Dynamic Chaos in Automatically Open on Orbit of Large-Dimension Folding Reflectors Space Mirror Antennas of a Truss Type, Executed as of Spatial Multibody Systems

17:10 - 17:30 Y. Gonthier, J. McPhee, Ch. Lange, J.C. Piedboeuf
A Regularized Contact Model with Asymmetric Damping and Dwell-Time Dependent Friction

Thursday , 27 June 2002

08:30 - 10:30 Lecture 7
F. Pfeiffer / K. Funk
Unilateral Multibody Dynamics

11:00 - 13:00 Lecture 8
V. Berbyuk
Control and Optimization of Semi-Passively Actuated Multibody Systems

Session 7 - Chairperson: H. Lankarani

14:30 - 14:50 B. Muth, P. Eberhard
Collision Detection and Administration for Many Colliding Bodies

14:50 - 15:10 I. Zanevskyy
Mechanical and Mathematical Modelling and Computer Simulation of Vibration and Impact Processes in the 'Man and Shooting Device' Systems

15:10 - 15:30 M. Anitescu
Solving Nonconvex Problems of Multi-Body Dynamics with Contact and Small Friction
by Sequential Convex Relaxation

15:30 - 15:50 V. N. Yazykov
Some Results of Wheel-Rail Contact Modelling

15:50 - 16:10 W.-S. Yoo, M.-G. Kim, K.-S. Kim
Modification of Road Profile to Compensate Tire Nonlinearity in Linear Tyre Model

Session 8 - Chairperson: A. A. Shabana

16:30 - 16:50 S. L. Pedersen, J. M. Hansen
A Novel Roller-Chain Drive Model Using Multibody Dynamics Analysis Tools

16:50 - 17:10 J. Fraczek
Kinematic Analysis of Mechanisms in the Neighbourhood of Singular Positions Using General Numerical Continuation Methods

17:10 - 17:30 D. Lefeber, J. Naudet, Z. Terze, F. Daerden
Forward Dynamics of Multibody Mechanisms Using an Efficient Algorithm Based on Canonical Momenta

Friday , 28 June 2002

08:30 - 10:30 Lecture 9
M. S. Pereira
Optimization of Rigid-Flexible Multibody Systems with Application to Vehicle Dynamics and Crashworthiness

11:00 - 13:00 Lecture 10
H. Lankarani
A Virtual Multibody and Finite Element Analysis Environment in the Field of Aerospace Crashworthiness

Session 9 - Chairperson: V. Berbyuk

14:30 - 14:50 K. E. Georgiev, T. Ivanova
Mechatronic Approach for Simulation of Robots and Walking Mashines

14:50 - 15:10 K. Gr. Kostadinov, G. V. Boiadjiev
Development of Impedance Control Method for Mechatronic Systems

15:10 - 15:30 S. F. Jatsun, A. S. Zaisev, S. M. Jatsun
Dynamics of Vibrating System with Active Control

15:30 - 15:50 O. Brüls, J.-C. Golinval
Simulation of an Active Control System in a Hot-Dip Galvanizing Line

Session 10 - Chairperson: P. Eberhard

16:30 - 16:50 R. Kovalev
Optimizing Multibody Systems: Some Implementations And Results

16:50 - 17:10 K. Belda, J. Böhm, M. Valášek
State-Space Generalized Predictive Control for Redundant Parallel Robots

Saturday, 29 June 2002

08:30 - 10:30 Lecture 11
E. Kreuzer
Multibody System Dynamics in Ocean Engineering

11:00 - 13:00 Lecture 12
J. McPhee
Graph-Theoretic Modelling of Multibody Systems

Sunday , 30 June 2002

08:30 - 19.00 Excursion - Trip

Monday , 1 July 2002

08:30 - 10:30 Lecture 13
E. J. Haug
Virtual Proving Ground Simulation for Highway Safety Research and Vehicle Design

11:00 - 13:00 Lecture 14
M. Valášek
Design of Nonlinear Control of Nonlinear Multibody Systems

Session 11 - Chairperson: E. Haug

14:30 - 14:50 G. Schupp
Simulation of Railway Vehicles: Necessities and Applications

14:50 - 15:10 A. Carrarini
Coupled Multibody-Aerodynamic Simulation of High-Speed Trains Manoeuvres

15:10 - 15:30 J. Pombo, J. Ambrósio
Development of a Roller Coaster Model

15:30 - 15:50 S.-S. Kim, M. Won, B. Sohn, K. Song, S. Jung
The Development of a Real-Time Multibody Vehicle Dynamics and Control Model for
a Low Cost Virtual Reality Vehicle Simulator: An Application to
Adaptive Cruise Control

15:50 - 16:10 P. P. Valentini, L. Vita
David – a Multibody Code to Simulate a Dynamic Virtual Dummy for Vibrational Comfort Analysis of Car Occupants

Session 12 - Chairperson: D. Bestle

16:30 - 16:50 J. Tobolár
Model Reduction Techniques for Vehicle Suspensions in Real-Time Applications

16:50 - 17:10 S. K. Agrawal, J. Yan, J. Franch
Dynamics and Control of a Vehicle with Expanding Wheels
Using Differential Flatness

17:10 - 17:30 K. Pathak
Model Reformulation in Dynamic Optimization -A Numerical Study-
Planning and Optimization

Tuesday , 2 July 2002

08:30 - 10:30 Lecture 15
W. Blajer
Geometrical Interpretation of Multibody Dynamics: Theory and Implementations

11:00 - 13:00 Lecture 16
D. Bestle
Optimization of Passive and Active Dynamic Systems

Session 13 - Chairperson: W. Blajer

14:30 - 14:50 M. A. Neto, J. Ambrósio
Stabilization Methods for the Integration of DAE in the presence
of Redundant Constraints

14:50 - 15:10 Z. Terze, D. Lefeber
MBS Time Integration-Projective Constraint Violation Stabilization Methods
on Manifolds

15:10 - 15:30 I. V. Boikov, A. I. Boikova
Stability of Solution of Differential Equations

15:30 - 15:50 A.Fuchs, M. Arnold
Efficient Corrector Iteration for Implicit Time Integration in Multibody Dynamics

15:50 - 16:10 F. Aghili, J.-C. Piedbœuf/ J. Kövecses
Simulation of Constrained Multibody Systems Based on Orthogonal Decomposition
of Generalized Coordinates

Session 14 - Chairperson: P. Nikravesh

16:30 - 16:50 D. Negrut
On the Issue of Iterative Linear Algorithms for the Multi-Threaded Simulation of Mechanical Systems Represented in Cartesian Coordinates

Wednesday, 3 July 2002

Session 15 - Chairperson: W. Schiehlen

08:30 - 08:50 A. Müller
Parallel Computing in the Context of Multibody System Dynamics

08:50 - 10:35 Lecture 17
E. Zahariev
Multibody System Contact Dynamics Simulation

11:00 - 12:45 Lecture 18
P. Eberhard
Contact Formulations for Finite Elements and Multibody Systems

12:45 - 13:00 Closing session

Participant List NATO Advance Study Institute Prague, 23.6.-3.7. 2002

surname	first name	university / institute	address	country	e-mail
DIRECTORS					
Schiehlen	Werner	University of Stuttgart	Institut B of Mechanics University of Stuttgart Pfaffenwaldring 9 Stuttgart	Germany	wos@mechb.uni-stuttgart.de
Valášek	Michael	CTU in Prague, Faculty of Mechanical Enginnering	Dept. of Mechanics Faculty of Mechanical Eng., CTU in Prague Karlovo nam 13 121 35 Praha 2	Czech Republic	valasek@fsik.cvut.cz
LECTURERS					
CANADA					
McPhee	John	University of Waterloo	Systems Design Engineering University of Waterloo 200 University Avenue West N2L 3G1 Waterloo, Ontario	Canada	mcphee@real.uwaterloo.ca
FRANCE					
Pascal	Madeleine	Université Pierre et Marie Curie	Université Pierre et Marie Curie, LMM Tour 66, Case 162 4, Place Jussieu 75252 Paris Cedex 05	France	pascal@lmm.jussieu.fr
GERMANY					
Bestle	Dieter	BTU Cottbus	Lehrstuhl fur Maschinendynamik BTU Cottbus Universitaetsplatz 3/4 03044 Cottbus	Germany	bestle@TU-Cottbus.De
Eberhard	Peter	University of Erlangen-Nuremberg	Institute of Applied Mechanics University of Erlangen-Nuremberg Egerlandstr. 5D 91058 Erlangen]	Germany	eberhard@ltm.uni-erlangen.de
Funk	Kilian	Technische Universitaet Muenchen	Technische Universitaet Muenchen Lehrstuhl fuer Angewandte Mechanik D-85747 Garching	Germany	funk@amm.mw.tu-muenchen.de
Kreuzer	Edwin	Technische Universitat Hamburg	Eissendorfer Str. 42 D-21073 Hamburg	Germany	kreuzer@tu-harburg.de
POLAND					
Blajer	Wojtiech	Technical University of Radomul. Krasickiego	Institute of Applied Mechanics Technical University of Radomul. Krasickiego 5426-600 Radom	Poland	wblajer@poczta.onet.pl
PORTUGAL					
Ambrosio	Jorge	IDMEC/IST, Instituto Superior Técnico	Av. Rovisco Pais 11096 Lisboa	Portugal	jorge@dem.ist.utl.pt
Pereira	Manuel	IDMEC / Instituto Superior Técnico	IDMEC / Instituto Superior Técnico Mechanical Engineering Department Av. Rovisco Pais 1096 Lisboa	Portugal	pcd3209@alfa.ist.utl.pt
USA					
Haug	Edward	NADS and Simulation Center	BlvdCoralville 2401 Oakdale IA 52241	USA	haug@nads-sc.uiowa.edu

Nikravesh	Parviz	University of Arizona	University of Arizona Dept. of Aerospace & Mechanical Engineering AZ 85721 Tucson	USA	pen@ame.arizona.edu
Lankarani	Hamid	Wichita State University	Mechanical Engineering Dept. Wichita State University KS 67260-0133 Wichita	USA	lankaran@me.twsu.edu
Shabana	Ahmed	2039 Engineering Research Facility	Dept of Mechanical Engng 2039 Engineering Research Facility 842 West Taylor Street Chicago, IL 60607-7022	USA	shabana@uic.edu
BULGARIA					
Zakhariev	Evtim	Institute of Mechanics, Bulgarian Academy of Sciences	Institute of Mechanics Bulgarian Academy of Sciences Acad. G. Bonchev Street, bl. 4 1113 Sofia	Bulgaria	evtimvz@bgcict.acad.bg
RUSSIA					
Chernousko	Felix	Institute for Problems in Mechanics, Russian Academy of Science	Institute for Problems in Mechanics Russian Academy of Science Pr. Vernadskogo 101 117526 Moscow	Russia	chern@ipmnet.ru
SWEDEN					
Berbyuk	Victor	Chalmers University of Technology	Mechanical Systems Division Department of Machine and Vehicle Systems Chalmers University of Technology SE-412 96 Gothenburg	Sweden	viktor.berbyuk@me.chalmers.se
STUDENTS					
NATO	**Countries**				
BELGIUM					
Brüls	Olivier	LTAS - Aérospatiale, Mécanique et Matériaux	Institut de Mécanique et de Génie Civil, Bât. B52 Chemin des Chevreuils, 1 B-4000 Liege	Belgium	o.bruls@ulg.ac.be
Lefeber	Dirk	University of Brussels, Faculty of Applied Science	University of Brussels, Faculty of Applied Science Pleinlaan 2 B-1050 Brussels	Belgium	dlefeber@vub.ac.be
Naudet	Joris	Free University Brussels (VUB)	Free University of Brussels Faculty of Applied Sciences Department of Mechanical Engineering Pleinlaan 2 1050 Brussels	Belgium	Joris.Naudet@vub.ac.be
CANADA					
Kovecses	Jozsef	Space Technologies, Canadian Space Agency	6767 route de l'Aéroport, Saint-Hubert Québec Canada J3Y 8Y9	Canada	Jozsef.Kovecses@space.gc.ca
Schmitke	Chad	University of Waterloo	Systems Design Engineering University of Waterloo 200 University Ave. W. Waterloo, Ontario N2L 3G1	Canada	schmitke@real.uwaterloo.ca
CZECH REP.					
Belda	Květoslav	Institute of Information Theory and Automation, AS	Pod vodárenskou věží 4 P.O. Box 18	Czech Republic	belda@utia.cas.cz

			182 08 Prague 8 - Libeň Czech Republic		
Beneš	Karel	MECAS	Uslavska 10 Plzen 32600	Czech Republic	karel.benes@mecasesi.cz
Hyncik	Ludek	New Technologies Research Centre in the West-Bohemia	Univerzitní 8 306 14 Plzeň	Czech Republic	hyncik3@ntc.zcu.cz
Šika	Zbynek	CTU in Prague, Faculty of Mechanical Enginnering	Dept. of Mechanics Faculty of Mechanical Eng., CTU in Prague Karlovo nam 13 121 35 Praha 2	Czech Republic	sika@felber.fsik.cvut.cz
Steinbauer	Pavel	CTU in Prague, Faculty of Mechanical Enginnering	Dept. of Mechanics Faculty of Mechanical Eng., CTU in Prague Karlovo nam 13 121 35 Praha 2	Czech Republic	steinbauer@fsik.cvut.cz
DENMARK					
Pedersen	Sine Leergaard	Technical University of Denmark	Nils Koppelsallé Bygn 404 DK - 2800 Kgs. Lyngby	Denmark	slp@mek.dtu.dk
FRANCE					
Dequidt	Antoine	Universiteacute de Valenciennes	ENSIMEV Le Mont Houy F-59313 VALENCIENNES CEDEX 9	France	antoine.dequidt@univ-valenciennes.fr
Taralova	Ina	Institut de Recherche en Communications et Cyberne	Ecole Centrale de Nantes 1 Rue de la Noe 44321 Nantes	France	ina.taralova@irccyn.ec-nantes.fr
GERMANY					
Carrarini	Antonio	DLR - Vehicle System Dynamics Group	P.O. Box 1116 D-82230 Wessling	Germany	antonio.carrarini@dlr.de
Eiber	Albrecht	Institute B of Mechanics	Pfaffenwaldring 9 70550 Stuttgart	Germany	ae@mechb.uni-stuttgart.de
Ellermann	Katrin	TU Hamburg-Harburg	TU Hamburg-Harburg Mechanik und Meerestechnik 21071 Hamburg	Germany	ellermann@tu-harburg.de
Fuchs	Andreas	DLR German Aerospace Center	Institute of Aeroelasticity Vehicle System Dynamics Group P.O.Box 1116, D-82230 Wessling	Germany	andreas.fuchs@dlr.de
Kuebler	Lars	Univ. Erlangen	Institute of Applied Mechanics, Univ. Erlangen Egerlandstr. 5 91058 Erlangen	Germany	kuebler@ltm.uni-erlangen.de
Lampariello	Roberto	DLR German Aerospace Center	Institute of Robotics P.O.Box 1116, D-82230 Wessling	Germany	roberto.lampariello@dlr.de
Mueller	Andreas	Institute of Mechatronic at TU Chemnitz	Institute of Mechatronics at the TU Chemnitz, Reichenhainer 5tr. 88, 09126 Chemnitz	Germany	Andreas.Mueller@ifm.tu-chemnitz.de
Muth	Beate	Institute B of Mechanics	University of Stuttgart Pfaffenwaldring 9 70550 Stuttgart	Germany	bm@mechb.uni-stuttgart.de
Schupp	Gunter	DLR	Deutsches Zentrum fuer Luft- und Raumfahrt Institut fuer Aeroelastik – Fahrzeug-Systemdynamik Postfach 11 16D-82230 Wessling	Germany	Gunter.Schupp@dlr.de
Tobolar	Jakub	DLR, Institute of Aeroelasticity	Deutsches Zentrum für Luft- und Raumfahrt	Germany	jakub.tobolar@dlr.de

			Institut für Aeroelastik P.O.Box 1116 D-82230 Weßling		
von Dombrowski	Stefan	DLR - German Aerospace Center	DLR Institute of Robotics and Mechatronics Research Center Oberpfaffenhofen D-82234 Weßling	Germany	Stefan.von.Dombrowski@dlr.de
ITALY					
Valentini	Pier Paolo	Università di Roma Tor Vergata	Dipartimento di Ingegneria Meccanica via del Politecnico, 1 00133 Roma	Italy	Valentini@ing.uniroma2.it
Vita	Leonardo	Università; degli Studi di Roma Tor Vergata	Dipartimento di Ingegneria Meccanica Via del Politecnico 1 00133 Roma	Italy	Vita@ing.uniroma2.it
NETHERLANDS					
Sciacovelli	Donato	ESA-ESTEC	Kepleerlaan 1, 2200AG Nordwijk	Netherlands	donato.Sciacovelli@esa.int
Schwab	Arend L.	Delft University of Technology	Mekelweg 2 NL 2628 CD Delft	Netherlands	a.l.schwab@wbmt.tudelft.nl
POLAND					
Czaplicki	Adam	Academy of Physical Education in Warsaw	Ul. Akademicka 2 21-500 Biala Podlaska	Poland	czaplicki@poczta.onet.pl
Fraczek	Janusz	Institute of Aeronautics and Applied Mechanics	Nowowiejska 24 00-665 Warsaw	Poland	jfraczek@meil.pw.edu.pl
Kolodziejczyk	Krzysztof	Technical University of Radom	Technical University of Radom Institute of Applied Mechanics ul. Krasickiego 54 26-600 Radom	Poland	krkolo@poczta.onet.pl
Wojtyra	Marek	Warsaw University of Technology	Institute of Aeronautics and Applied Mechanics Warsaw University of Technology ul. Nowowiejska 24 00-665 Warszawa	Poland	mwojtyra@meil.pw.edu.pl
PORTUGAL					
Dias	Joao	IDMEC/IST	Av. Rovisco Pais	Portugal	jdias@dem.ist.utl.pt
Flores Fernandes	Joao Paulo	University of Minho	Universidade do Minho Departamento de Engenharia Mecanica Campus de Azurem 4800-058 Guimaraes	Portugal	pflores@dem.uminho.pt
Neto	Maria Augusta	Universidade de Coimbra	Dep. de Eng-a Mecanica Pinhal de Marrocos (polo II) 3030 Coimbra	Portugal	augusta.neto@dem.uc.pt
Pombo	Joao	IDMEC/Instituto Superior Técnico	IDMEC/Instituto Superior Técnico Av. Rovisco Pais 1049-001 Lisboa	Portugal	jpombo@bigfoot.com
Tavares da Silva	Miguel	IDMEC/Instituto Superior Técnico	Instituto Superior Técnico Departamento de Engenharia Mecânica Av. Rovisco Pais 1049-001 Lisboa	Portugal	pcms@alfa.ist.utl.pt
SPAIN					
Escalona	José L.	University of Seville	Department of Mechanical Engineering University of Seville Camino de los Descubrimientos s/n 41092 Seville	Spain	escalona@us.es

Valverde	Juan	University of Seville	Department of Mechanical Engineering University of Seville Camino de los Descubrimientos s/n 41092 Seville	Spain	jvalverde@us.es
TURKEY					
Camlibel	M. Kanat	University of Groningen	University of Groningen Dept. of Mathematics P.O. Box 800 9700 AV Groningen	Turkey	k.camlibel@math.rug.nl
Kalkan	Nevin	University of Ystanbul	University of Ystanbul, Faculty of Science, Physics Department Vezneciler - YSTANBUL	Turkey	nevinkalkan@hotmail.com
UK					
Wray	Will	Mathengine Plc.	60 St. Aldates, Oxford OX1 1ST	United Kingdom	will.wray@mathengine.com
USA					
Agrawal	Sunil	University of Delaware	222 Spencer Laboratory Department of Mechanical Engineering University of Delaware Newark, DE 19716	USA	agrawal@me.udel.edu
Anitescu	Mihai	University of Pittsburgh	University of Pittsburgh, Department of Mathematics, Thackeray 301, Pittsburgh, PA 15213. USA	USA	anitescu@math.pitt.edu
Negrut	Dan	Mechanical Dynamics, Inc.	2300 Traverwood DriveAnn Arbor Mi-48105	USA	dnegr@adams.com
Pathak	Kaustubh	University of Delaware	134 Spencer, Mechanical Systems Lab Dept. of Mechanical Engineering University of Delaware Newark, DE 19716	USA	pathak@me.udel.edu
Yan	Jin	University of Delaware	134 Spencer, Mechanical Systems Lab Department of Mechanical Engineering University of Delaware, Newark, DE 19716	USA	yanj@me.udel.edu
Eligible	**Partner**	**Countries**			
BULGARIA					
Georgiev	Krassimir	BAS	Mechatronic systems department Institute of Mechanics, BAS acad.G.Bonchev street, blok 4 1113 Sofia	Bulgaria	kris@imbm.bas.bg
Kostadinov	Kostadin	Bulgarian Academy of Sciences	Acad. G. Bonchev St., Block 4 1113 Sofia	Bulgaria	kostadinov@imbm.bas.bg
Vassileva	Daniela	Institute of Mechanics	Acad. G. Bonchev St., Block 4 1113 Sofia	Bulgaria	Daniela@imbm.bas.bg
CROATIA					
Terze	Zdravko	University of Zagreb, Faculty of Mech. Eng. and Na	University of Zagreb Faculty of Mech. Eng. and Na Ivana Lurica 5, 10000 Zagreb	Croatia	zdravko.terze@fsb.hr
ROMANIA					
Lache	Simona	'TRANSILVANIA' UNIVERSITY OF BRASOV	29 B-DUL EROILOR RO-2200 BRASOV	Romania	slache@unitbv.ro
Micu	Ioan	University TRANSILVANIA of Brasov	B-dul Eroilor,29, 2200, Brasov	Romania	mtofan@unitbv.ro

Stroe	Ion	POLITEHNICA University of Bucharest	POLITEHNICA University of Bucharest Splaiul Independentei 313 Bucharest RO-77206	Romania	ion.stroe@rosa.ro
Talaba	Doru	University Transilvania of Brasov	Product Design and Robotics Department Bd. Eroilor 29 Brasov RO-2200	Romania	talaba@unitbv.ro
Vlase	Sorin	University TRANSILVANIA of Brasov	b-dul Eroilor, 29 2200, Brasov	Romania	svlase@yahoo.com
RUSSIA					
Boikov	Ilia	Penza State University	Home: Kirov str, 71-58, 440600, Penza Office: Krasnay, 40, 440026, Penza	Russia	boikov@diamond.stup.ac.ru
Boikova	Alla	Penza State University	Home: Volodarsky, 68-15, 440600, Penza Office: Krasnay str, 40 ,440026, Penza	Russia	allaboikova@sura.ru
Dmitrotchenko	Oleg	Bryansk State Technical University	Dept. of Applied Mechanics Bryansk State Technical University bulv. im. 50-letiya Oktyabrya 7 241035 Bryansk	Russia	don@bitmcnit.bryansk.su
Figurina	Tatiana	Institute for Problems in Mechanics, RAS	101-1 prospect Vernadskogo Moscow, 117526	Russia	figurina@ipmnet.ru
Jatsun	Sergey	Kursk State Technical University	50 Let Oktyabrya 94, room 218 Kursk 305040	Russia	jatsun@kursknet.ru
Kolesnikov	Maxim	Moscow State Institute of Steel and Alloys	MKR.VOSTOCHNY 17-31 STARY OSKOL 309502 BELGORODSKAYA OBL.	Russia	maxkol@oskolnet.ru
Kovalev	Roman	Bryansk State Technical University	Bryansk State Technical University Dept. of Applied Mechanics bulv. 50-let Oktyabrya, 7 Bryansk, 241035	Russia	kovalyov@bitmcnit.bryansk.su
Pogorelov	Dmitri	Bryansk State Technical University	of Applied Mechanics Bryansk State Technical University b. 50 let Oktyabrya 7241035 Bryansk	Russia	pogorelov@bitmcnit.bryansk.ru
Sayapin	Sergey	Mechanical Engineering Research Institute Russian	Flat 188 75 Marx St.	Russia	vibris@interface.ru
Yazykov	Vladislav	Bryansk State Technical University	Dept. of Applied Mechanics, State Technical University, bulv. im. 50-letiya Oktyabrya, 7, 241035, Bryansk	Russia	mechan@bitmcnit.bryansk.su
SLOVAK REP.					
Polakova	Zuzana	KAME, Sj.F. KOSICE	Gocovo 179, 049 24, okr. ROZNAVA	Slovakia	polakzuzu@yahoo.co.uk
UKRAINE					
Zanevskyy	Ihor	Lviv State Institute of Physical Culture	Kostyushko str. 11 Lviv 79000 Ukraine	Ukraine	izanevskyy@hotmail.com
Non-Eligible	**Partner**	**Countries**			
FINLAND					
Mikkola	Aki	Lappeenranta University of Technology	Lappeenranta University of Technology Department of Mechanical Engineering PL 20, 53851 Lappeenranta	Finland	mikkola@lut.fi

SWEDEN					
Lidberg	Mathias	Chalmers University of Technology	Department of Applied Mechanics Chalmers University of Technology SE-412 96 Gothenburg	Sweden	mathias.lidberg@me.chalmers.se

MULTIBODY SYSTEMS WITH HOLONOMIC AND NONHOLONOMIC CONSTRAINTS

Kinetics

I. STROE
"Politehnica" University of Bucharest,
77206, Bucharest, Romania
ion.stroe@rosa.ro

Using Lagrange equations for holonomic and non-holonomic systems the motion of systems of rigid bodies is studied in this paper.
General problem of kinematics of systems is presented in the first part of the paper.
The motion of systems of rigid bodies with constraints is studied in the last part of the paper. Motions of rigid bodies with articulation joints are analyzed. Problems of kinematics are solved for constraints expressed by coordinates. Translation conditions and rotation conditions are analyzed.
When the motion of a system of bodies which compose a large orbital station is described with respect to reference frames having origin in the center of attractive body (Earth) the problem of integration of motion equations presents some difficulties, because some coordinates (like vector radii) have very great values, and others (like distances between bodies) have very small values. Some difficulties can be avoided if relative motion of the system is studied with respect to a reference frame with known motion. Relative motion study isn't impose by integration considerations, this is impose by practical aspects.
The problem of kinematics for systems of bodies are solved using analyzes of coupling mechanism under the aspect of number of degrees-of -freedom. The motion in central gravitational field is studied with respect a movable reference frame with origin on a circular orbit. The problem of dynamics of bodies system is solved using Lagrange equations of motion with multipliers and constraints. The models and the elaborated method allow to solve a large number of problems of bodies systems dynamics in gravitational field.

1. Kinematics of systems of rigid bodies

Let two bodies (*i*) and (*j*) be with constrained motions by a coupling mechanism which is made precise by points O_i, O_j (fig.1).
The motion of the body (*i*) with respect the inertial reference frame $O_0x_0y_0z_0$ is determined by position vector of mass center $\overline{O_0C_i}$ and by matrix $\left[A_{i0}\right]$ which gives the attitude of $C_ix_iy_iz_i$ triedron, jointed with (*i*) body, with respect $O_0x_0y_0z_0$ reference frame.
In the same way are defined position vector $\overline{O_0C_j}$ and matrix $\left[A_{j0}\right]$ for the body (*j*).
Each body, (*i*) or (*j*), has 6 degrees-of- freedom, when it is a free body. The number of degrees-of- freedom is reduced by the number of constrains which are imposed by coupling mechanism.
If the general motion of bodies (*i*) and (*j*) with respect the inertial reference frame $O_0x_0y_0z_0$ are known, then the relative motion of the body (*i*) with respect (*j*) can be determined by vector

$$\overline{O_jO_i} = \left(\overline{O_0C_i} + \overline{C_iO_i}\right) - \left(\overline{O_0C_j} + \overline{C_jO_j}\right) \qquad (1)$$

and by matrix $\left[A_{ij}\right]$ which gives the attitude of (i) body with respect (j) body,

$$\left[A_{ij}\right] = \left[A_{i0}\right]\left[A_{j0}\right]^T \qquad (2)$$

The matrix $\left[A_{i0}\right]$ allows expressing unit vectors of $C_ix_iy_iz_i$ trihedron with respect unit vectors of $O_0x_0y_0z_0$ trihedron,

$$\begin{Bmatrix} \overline{i_i} \\ \overline{j_i} \\ \overline{k_i} \end{Bmatrix} = \left[A_{i0}\right] \begin{Bmatrix} \overline{i_0} \\ \overline{j_0} \\ \overline{k_0} \end{Bmatrix} . \qquad (3)$$

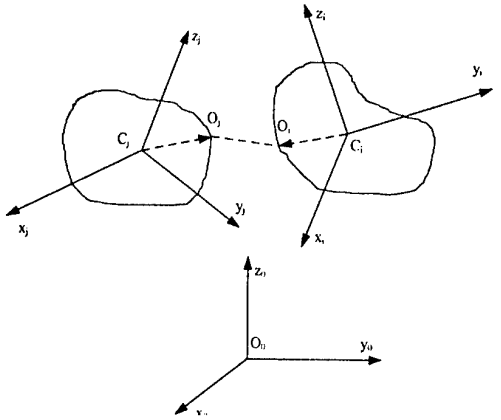


Fig.1 System of rigid bodies

For unit vectors of $C_ix_iy_iz_i$ trihedron the bellow relation can be written,

$$\begin{Bmatrix} \overline{i_j} \\ \overline{j_j} \\ \overline{k_j} \end{Bmatrix} = \left[A_{j0}\right] \begin{Bmatrix} \overline{i_0} \\ \overline{j_0} \\ \overline{k_0} \end{Bmatrix} \qquad (4)$$

The attitude of (i) body with respect to (j) body is given by matrix $\left[A_{ij}\right]$, with relations

$$\begin{Bmatrix} \overline{i_i} \\ \overline{j_i} \\ \overline{k_i} \end{Bmatrix} = \left[A_{ij}\right] \begin{Bmatrix} \overline{i_j} \\ \overline{j_j} \\ \overline{k_j} \end{Bmatrix} \qquad (5)$$

and the attitude of (j) body with respect (i) is given by $\left[A_{ji}\right]$ matrix from relations

$$\begin{Bmatrix} \overline{i_j} \\ \overline{j_j} \\ \overline{k_j} \end{Bmatrix} = \left[A_{ji}\right] \begin{Bmatrix} \overline{i_i} \\ \overline{j_i} \\ \overline{k_i} \end{Bmatrix} \qquad (6)$$

From (5) and (6) relations it follows

$$\left[A_{ij}\right]=\left[A_{ji}\right]^T \tag{7}$$

If (3) relation is multiplied to the left with $\left[A_{i0}\right]^T$ and (4) is multiplied to the left with $\left[A_{j0}\right]^T$ and the obtained results are compared, the equality

$$\left[A_{i0}\right]^T\begin{Bmatrix}\overline{i_i}\\ \overline{j_i}\\ \overline{k_i}\end{Bmatrix}=\left[A_{j0}\right]^T\begin{Bmatrix}\overline{i_j}\\ \overline{j_j}\\ \overline{k_j}\end{Bmatrix} \tag{8}$$

results, from where, by multiplying to the left with $\left[A_{i0}\right]$, the bellow relation is obtained

$$\begin{Bmatrix}\overline{i_i}\\ \overline{j_i}\\ \overline{k_i}\end{Bmatrix}=\left[A_{i0}\right]\left[A_{j0}\right]^T\begin{Bmatrix}\overline{i_j}\\ \overline{j_j}\\ \overline{k_j}\end{Bmatrix} \tag{9}$$

From (5) and (9) relations the (1) relation is obtained, which is used to compute the matrix $\left[A_{ij}\right]$, if the matrices $\left[A_{i0}\right]$ and $\left[A_{j0}\right]$ are known. Terms of $\left[A_{i0}\right]$ and $\left[A_{j0}\right]$ matrices depend of attitude angles of (*i*) body and (*j*) body with respect to inertial reference frame. If the orientation of (*i*) body with respect to inertial reference frame is made precise by φ_{1i}, φ_{2i}, φ_{3I} angles, which correspond to the 1-2-3 sequence of rotations with respect to a parallel reference frame with inertial reference frame $O_0x_0y_0z_0$, than the bellow matrix

$$\left[A_{i0}\right]=\begin{bmatrix} c_{2i}c_{3i} & s_{1i}s_{2i}c_{3i}+s_{3i}c_{1i} & -c_{1i}s_{2i}c_{3i}+s_{1i}s_{3i}\\ -c_{2i}s_{3i} & -s_{1i}s_{2i}s_{3i}+c_{1i}c_{3i} & c_{1i}s_{2i}s_{3i}+s_{1i}c_{3i}\\ s_{2i} & -s_{1i}c_{2i} & c_{1i}c_{2i}\end{bmatrix} \tag{10}$$

and angular velocity

$$\left\{\omega_{i0}\right\}=\begin{bmatrix} c_{2i}c_{3i} & s_{3i} & 0\\ -c_{2i}s_{3i} & c_{3i} & 0\\ s_{2i} & 0 & 1\end{bmatrix}\begin{Bmatrix}\dot{\varphi}_{1i}\\ \dot{\varphi}_{2i}\\ \dot{\varphi}_{3i}\end{Bmatrix} \tag{11}$$

are obtained.

In the above relations notations of the following form were used:

$$s_{1i}=\sin\varphi_{1i}\ ,\quad c_{1i}=\cos\varphi_{1i} \tag{12}$$

When constraints are functions of coordinates the motion of systems of rigid bodies can be studied with Lagrange equations for holonomic systems with dependent variables. Coupling mechanisms between (*i*) body and (*j*) body imposes restrictions on relative motion of (*i*) body with respect to (*j*).

Bellow some simple coupling mechanisms for which constraints can be expressed with functions of coordinates or with functions of velocities are analyzed.

1.1 CONSTRAINTS EXPRESSED BY COORDINATES

1.1.1 "Free" linkage

When the coupling mechanism doesn't impose restrictions coordinates which are describing relative motion (displacements and rotations) number of constraints is zero. Each body ((*i*) or (*j*) has 6 degrees-of- freedom and the motion is studied considering two free bodies, despite of the coupling mechanism, which permits, translations with respect three directions and

rotations about three axes. The case of "free" linkage is a limit case and it presents the importance only for the case in which a particular coupling mechanism becomes a "free" linkage. Like an example can be considered the case of tethered bodies for particular situation of zero tension in the cable.

1.1.2 Fixed linkage

When the relative motion of (i) body with respect to (j) body has zero degrees-of-freedom the system of two bodies becomes a rigid one and it has

$$6 \quad + \quad 6 \quad - \quad 6 \quad = 6$$

$$(i)body \quad (j)body \quad constraints$$

degrees-of-freedom.

Relative displacement condition, in vectorial form is

$$\overline{O_jO_i} = \bar{0}, \tag{13}$$

and conditions of invariable relative orientation are:

$$\begin{aligned} &\bar{i}_i \cdot \bar{i}_j = \left(\bar{i}_i \cdot \bar{i}_j\right)_0, \\ &\bar{j}_i \cdot \bar{j}_j = \left(\bar{j}_i \cdot \bar{j}_j\right)_0, \\ &\bar{k}_i \cdot \bar{k}_j = \left(\bar{k}_i \cdot \bar{k}_j\right)_0 \end{aligned} \tag{14}$$

Index "$_0$" from right part of above relations corresponds to initial moment and it shows that inner products from left side are constants.

If (5) relation is written in the form

$$\begin{Bmatrix} \bar{i}_i \\ \bar{j}_i \\ \bar{k}_i \end{Bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}_{i,j} \begin{Bmatrix} \bar{i}_j \\ \bar{j}_j \\ \bar{k}_j \end{Bmatrix}, \tag{15}$$

(14) relations become:

$$a_{11}^{ij} = \left(a_{11}^{ij}\right)_0, \quad a_{22}^{ij} = \left(a_{22}^{ij}\right)_0, \quad a_{33}^{ij} = \left(a_{33}^{ij}\right)_0. \tag{16}$$

1.1.3 Spherical joint

Spherical joint reduces the number of degrees-freedom with three units. Vectorial form of constraint is (13) condition.

1.1.4 Linkage of translation

When the coupling mechanism allows translations in three some directions the number of degrees-freedom is reduced with three units and constraints are of the (16) form.

1.1.5 Connection with flexible cable

Coupling mechanism with flexible cable reduces the number of degrees-of-freedom with one unit. The distance between points of connection of flexible cable is a constant one, and conditions is

$$\left|\overline{O_jO_i}\right| = \left|\overline{O_jO_i}\right|_0 \tag{17}$$

When constraints are expressed by velocities (velocities of translations or angular velocities) the motion is described with Lagrange equations for non-holonomic systems. Coupling mechanism can be analyzed from the point of view of allowed mobility.

1.2 CONSTRAINTS EXPRESSED BY VELOCITIES

1.3.1. Translation conditions

If the coupling mechanism allows translations in three some directions, the number of constraints which correspond to translations is zero.

If the coupling mechanism allows translations in two directions of vectors $\overline{t_{j1}}\left(t_{j1x},t_{j1y},t_{j1z}\right)$ and $\overline{t_{j2}}\left(t_{j2x},t_{j2y},t_{j2z}\right)$ with components in the system of (j) body, than the constraint is expressed by inner product

$$\left(\overline{v_{Oi}}-\overline{v_{Oj}}\right)\cdot\left(\overline{t_{j1}}\times\overline{t_{j2}}\right)=0 \qquad (18)$$

1.3.2. Rotation conditions

When the coupling mechanism allows rotations with respect three some directions, the number of constrains is zero.

If the coupling mechanism allows rotations with respect two determined directions by vectors $\overline{r_{j1}}\left(r_{j1x},r_{j1y},r_{j1z}\right)$ and $\overline{r_{j2}}\left(r_{j2x},r_{j2y},r_{j2z}\right)$, which are expressed with components in (j) body reference frame, condition

$$\left(\overline{\omega_{i,0}}-\overline{\omega_{j,0}}\right)\cdot\left(\overline{r_{j1}}\times\overline{r_{j2}}\right)=0 \qquad (19)$$

can be written.If the coupling mechanism allows one rotation with respect the determined direction by vector $\overline{r_j}\left(r_{jx},r_{jy},r_{jz}\right)$, which is expressed with components in (j) body reference frame, than two scalar conditions which are included in vectorial form

$$\overline{\omega_{i,0}}-\overline{\omega_{j,0}}=\lambda_r\overline{r_j}, \qquad (20)$$

can be written, or in matric form,

$$\left[A_{ji}\right]\left\{\omega_{i,0}\right\}-\left\{\omega_{j,0}\right\}=\lambda_r\left\{r_j\right\} \qquad (21)$$

2. Motion equations

When the motion of a system of bodies which compose a large orbital station is described with reference frames having origin in the center of attractive body (Earth) the problem of integration of motion equations presents some difficulties, because some coordinates (like vector radii) have very great values, and others (like distances between bodies) have very small values. Some difficulties can be avoided if relative motion of the system is studied with respect to a reference frame with known motion. Relative motion study isn't impose by integration considerations, this is impose by practical aspects.

For a non-holonomic rheonomic system Lagrange equations for h coordinates

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial E}{\partial\dot{q}_k}\right)-\frac{\partial E}{\partial q_k}=Q_k+\sum_{i=1}^{p}\lambda_i a_{ik},\left(k=1,2,\cdots,h\right) \qquad (22)$$

are completed with constraints,

$$\sum_{k=1}^{h} a_{ik}\,\mathrm{d}q_k + b_i\mathrm{d}t = 0\,,\ (i = 1,2,\cdots p)\,. \qquad (23)$$

By solving of the system of h (22) equations and of (23) equations, q_k coordinates and λ_i multipliers are found.

From (22) equations for holonomic system can be obtained by replacement of a_{ik} functions.

In the case of a holonomic system constraints are of the form

$$\Phi_i(q_1,\cdots,q_h,t) = 0\,,\ (i = 1,2,\cdots p) \qquad (24)$$

and differential form is obtained,

$$\sum_{k=1}^{h} \frac{\partial \Phi_i}{\partial q_k}\,dq_k + b_i\,dt = 0\,,(i = 1,2,\cdots p) \qquad (25)$$

From (25) and (23) it follows

$$a_{ik} = \frac{\partial \Phi_i}{\partial q_k}, \qquad (26)$$

and (22) equations become

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial E}{\partial \dot{q}_k}\right) - \frac{\partial E}{\partial q_k} = Q_k + \frac{\partial}{\partial q_k}\sum_{i=1}^{p}\lambda_i\Phi_i\,,\ (k = 1,2,\cdots,h)\,. \qquad (27)$$

If the function

$$U_\phi = \sum_{i=1}^{p}\lambda_i\Phi_i \qquad (28)$$

is introduced, than equations (27) can be written in the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial E}{\partial \dot{q}_k}\right) - \frac{\partial E}{\partial q_k} = Q_k + \frac{\partial U_\Phi}{\partial q_k}\,,\ (k = 1,2,\cdots,h) \qquad (29)$$

From the h above equations and p constraints (24) functions which correspond to h generalized coordinates, q_k and to p multipliers λ_i are determined.

In the figure 2 two jointed bodies by a hinge are presented. The two bodies are situated in central gravitational field.

Plane relative motion is described by ρ_1, θ_1, φ_{31},ρ_2, θ_2, φ_{32}, coordinates.

Constraint

$$\overline{O_1O_2} = \bar{0} \qquad (30)$$

reduces the number of degrees-of-freedom from 6 to 4.

Relation (26) can be write in the form

$$\overline{CM_1} + \overline{M_1O_1} - \left(\overline{CM_2} + \overline{M_2O_2}\right) = \bar{0} \qquad (31)$$

and constraints are obtained using components of vectors from (31) on axes of Cx_cy_c reference frame:

$$\begin{aligned}\phi_1 &\equiv \rho_1\cos\theta_1 - \rho_2\cos\theta_2 + a_1\cos(\theta_1+\varphi_{31}) + b_2\cos(\theta_2+\varphi_{32}) = 0\\ \phi_2 &\equiv \rho_1\sin\theta_1 - \rho_2\sin\theta_2 + a_1\sin(\theta_1+\varphi_{31}) + b_2\sin(\theta_2+\varphi_{32}) = 0\end{aligned} \qquad (32)$$

From equations (29) and the above constraints equations of motion for the system of two jointed bodies by one hinge are obtained.

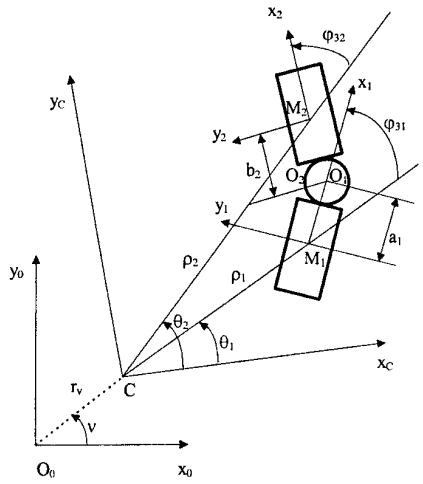


Fig.2 System of hinged bodies

3. Conclusions

The problem of kinematics for systems of bodies are solved using analyzes of coupling mechanism under the aspect of number of degrees-of -freedom.
The motion in central gravitational field is studied with respect a movable reference frame with origin on a circular orbit.
The problem of dynamics of bodies system is solved using Lagrange equations of motion with multipliers and constraints.
Models and elaborated method allow solving of a great number of problems of bodies systems dynamics.

4. References

1. Brouwer, D.,Clemence, G.M., *Methods of Celestial Mechanics*, Academic Press, New York and London, 1961.
2. Nita, M.M., *Teoria zborului spatial*, Ed. Academiei, Bucuresti, 1973.
3. Stroe, I., Prunariu, D.D., Piso, M.I., Attitude Control by the Change of Bodies Relative Position, *The 50th IAF Congress,* International Astronautical Federation, Amsterdam, *The Netherlands*, 4-8 Oct. 1999.
4. Stroe, I., Prunariu, D.D., Piso, M.I., Manciu, G.V., Dynamics of Large Objects Removal Systems, *Third European Conference on Space Debris*, ESOC, Darmstadt, Germany, 19-21 March 2001.
5. Stroe, I., Kinetics of Multibody Systems in Gravitational Field, *The Eight IFToMM International Symposium on Theory of Machines and Mechanisms*, Bucharest-ROMANIA, August 28-September 1, 2001,vol. II, pp.309-314.
6.Voinea, R., Stroe, I., A General Method for Kinematics Pairs Synthesis, *The Scientific Journal of IFToMM, Mech. Mach. Theory*, Pergamon, Elsevier Science Ltd.,Vol. 30, 3, 1995.
7. Voinea, R., Stroe, I., Predoi,M.V., *Technical Mechanics*, Geometry Balkan Press, Bucharest, 1996.
8. Voinea, R., Stroe, I., *Introducere în teoria sistemelor dinamice*, Editura Academiei Române, Bucuresti, 2000.

A PARTICLE MODEL FOR MECHANICAL SYSTEMS SIMULATION

A Model Based Overview of Multibody Systems Formulations

D. TALABA
University Transilvania of Brasov
Bd Eroilor 29, 2200-Brasov, Romania

Abstract
This paper presents a model-based overview of the formalisms for the simulation of the mechanical systems. This approach provides a clear background for any formulation including specific formulas for mobility computation and finally a proper evaluation of the potential for each of them. A full multi-particle model for the systems of interconnected rigids is also presented.

1. Introduction

In recent years, multibody analysis computer packages became a usual tool in industry, research and development areas. The commercially available codes include nowadays a large range of facilities allowing simulation of sophisticated experiments with virtual prototypes of mechanical systems (mechanisms). The cutting edge research in this field is currently aiming towards developing new modelling and simulation facilities related on one hand to including into the analytical formalisms complex non-linearity like flexibility of the bodies, friction modelling etc and on the other hand to the increasing of the computing speed in order to enable the real time simulation.

A large number of formalisms have been conceived and implemented in the various computer codes [7]. Some classifications of the methods utilized are taking into account the principle used for the dynamic formulation establishing two main kind of formalisms: Eulerian and Lagrangian. Usually, the two categories of methods use different sets of generalized coordinates and subsequently different methodologies for the kinematics formulation. Other classifications taken into account the type of implementation (i.e. symbolic or numerical implementation). This paper presents an overview of the formalisms from the model type viewpoint of the mechanical systems. This classification allows a synthetic picture the various methods (models) and subsequently an evaluation of the potential for each of them. Each formalism is based on a representation (model) of the physical system, from which all theoretical developments are derived. As resulting from the literature [2,3,4,5,6,7,8], two main representations have been assumed for the development of various methods and dynamic formalisms: the kinematic chain model and the multibody system model. Finally, the multi-particle model will be developed.

2. The kinematic chain model

According to this model, the mechanism is represented by a chain of bodies and interconnecting joints with the role of transmitting and transforming the motion (fig.1,a). The kinematic chain may be serial (open loop) or parallel (closed loop) and its structure is usually represented by a graph (fig.1,b), which allows automatic identification of the independent loops. This model has been implicitly assumed by some authors [6,8], being very popular especially in robotics.

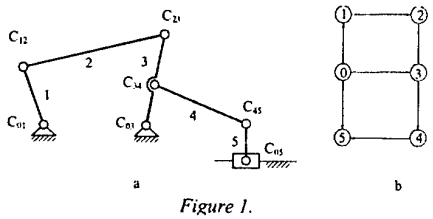


Figure 1.

For the serial mechanisms, the terminal body is cumulating the degrees of freedom of the preceding joints (fig.2), the structure mobility relationship being thus

$$M = \sum f_i. \quad (1)$$

In case of the parallel structures, the mobility is calculated in two steps: first the mechanism is converted into a serial structure by cutting a body in each closed loop. In this way, the mobility can be calculated with the relation (1). As the number of bodies was artificially increased (fig.3) one have to subtract the degrees of freedom number that vanish when the original elements are re-constituted as in the initial structure. In this way the mobility relation become:

$$M = \sum f_i - kS, \qquad (2)$$

in which k is the number of the closed loops and S – the motion dimension (S=3 for planar mechanisms and S=6 for spatial mechanisms). In some cases the same mechanism could include both planar and spatial loops at the same time. Therefore a more general formula is:

$$M = \sum f_i - \sum S_k, \qquad (3)$$

in which S_k is the motion dimension of the loop k.

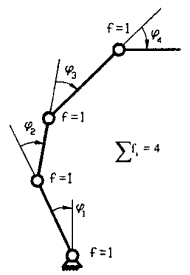


Figure 2.

The term $\sum S_k$ represents the number of constraints due to the closed loops. These constraints provide the same number of algebraic equations, which, together with the driving motions equations equal the dof allowed by the joints ($\sum f_i$), according to the relationship:

$$\sum f_i = M + \sum S_k . \qquad (4)$$

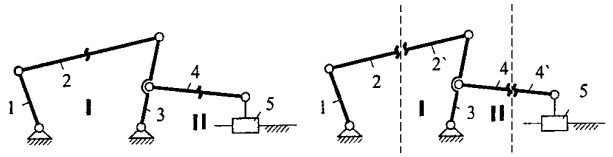


Figure 3.

$2 \equiv 2' \Rightarrow S_I = 3$
$4 \equiv 4' \Rightarrow S_{II} = 3$
$\sum f_i = 2+3+2 = 7$
$M = 7-6 = 1$

For the kinematics formulation, the motion of the mechanism modelled as a kinematic chain is characterized by a set of $\sum f_i$ kinematic equations.

For example for the mechanism from figure 3, $\sum f_i = 7$, which means the model has 7 generalized coordinates (the 7 articulations variables) and the geometric model includes 7 equations: 2×3=6 of them correspond to the closure conditions of loops I and II and one to the driving motion. According to the various formalisms, those equations can be written through various methods, most usually with Hartenberg-Denavit 4×4 operators. The set of 7 geometric equations constitutes a non-linear system of equations from which the values of the generalized coordinates can be obtained by numerical solving (Newton-Raphson). For the other kinematics equations formulation, recursive methods are well known from the literature [6,8], resulting in 7 velocity equations and 7 acceleration equations written in matricial form as:

$$\Phi(q,t) = 0, \quad \dot{\Phi}(q,t) = 0, \quad \ddot{\Phi}(q,t) = 0. \qquad (5)$$

For the dynamic analysis, the Newton-Euler formalism is usually involved, which means in principle writing for each body the equilibrium equations. In 2D space, for each body, 3 equilibrium equations can be written resulting finally $3n_c$ dynamic equations:

$$m\ddot{q} = Q + R, \qquad (6)$$

in which Q stands for the exterior forces and R the joint reactions.

These equations introduce as further unknowns the joint reactions, which are in number of $3n_j - \sum f_i$, where n_j is the number of joints. In this way, the differential algebraic equation (DAE) system is obtained as:

$$\begin{cases} \ddot{\Phi}(q,t) = 0, \\ m\ddot{q} = Q + R. \end{cases} \qquad (7)$$

For the sample mechanism in figure 3, out of the 6 acceleration equations, 15 dynamic equations can be written for the 5 bodies, introducing further 14 unknown reactions forces. The DAE system includes 6+15=21 equations with 21 unknowns: 7 generalized accelerations and 14 reactions forces that can be obtained by numerical integration.

3. The multibody system (MBS) model

According to this model, the mechanism is represented as a *collection of bodies,* the motion of which is subject to a set of absolute and relative constraints. Many authors have implicitly assumed this model since 1977 [6,4]. The mobility of the mechanism is obtained by cumulating all body's degrees of freedom considered as free bodies, from which the number of joint constraints is subtracted (Gruebler's formula):

$$M = 6 \cdot n - \Sigma i \cdot C_i \qquad (8)$$

in which S is the motion dimension of the space (S=6 for the spatial case and S=3 for the planar case), n is the number of the mechanism mobile bodies and C_i – the number of joints of class i (the class of a joint is given by the number of constraints introduced).

For illustration, the planar mechanism given in figure 3 have 5 bodies and 7 joints with two geometric constraints each, that is $M = 3 \cdot 5 - 7 \cdot 2 = 1$. In the kinematics case this mobility corresponds to the driving motion.

Each body is associated with a Body Reference Frame (BRF) characterized by three generalized coordinates (the origin coordinates and the orientation angle with respect to the Global Reference Frame – GRF), the mechanism position being characterized by $S \cdot n = 3 \cdot 5 = 15$ generalized coordinates [3].

Knowing the mechanism position is equivalent with knowing the BRF space position and orientation given in general by 6 coordinates (three origin coordinates and three orientation coordinates for each BRF). The total number of generalized coordinates is 6·n and the generalized coordinates vector is:

$$[q] = [x_1\ y_1\ z_1\ \varphi_1\ \psi_1\ \theta_1\ x_2\ y_2\ z_2\ \ldots\ldots\ x_n\ y_n\ z_n\ \varphi_n\ \psi_n\ \theta_n]^T \qquad (9)$$

or

$$[q] = [q_1\ q_2\ \ldots q_{6n}]^T \qquad (10)$$

Not all the coordinates are independent because of the geometrical constraints introduced by the joints. Each constraint is represented by a geometric condition written mathematically as an algebraic equation linking the generalized coordinates of the adjacent bodies. For example a tri-mobile joint (f=3, c=3) introduces three algebraic equations, a mono-mobile joint five algebraic equations etc. In total, the number of equations for all joints is $\Sigma i \cdot C_i$, where C_i is the numbers of joints of class i.

Consequently, the number of independent generalized coordinates (i.e. which can not be calculated from the constraint equations) equals the mechanism mobility,

$$N_{qi} = M = 6 \cdot n - \Sigma i \cdot C_i \,. \qquad (11)$$

The velocity and acceleration equations are derived generally by differentiation with respect to time of the position equations yielding relations with the expression (5).

The motion of the mechanism is cinematically determined when each independent generalized coordinate corresponds to a driving motion expressed by another algebraic equation.

For this kind of model, most usually the dynamic formulation includes 6·n differential equations with the general form

$$m\ddot{q} + J^T \lambda = Q_{ex}\,, \qquad (12)$$

where J is the constraints Jacobian, λ the Lagrange multipliers vector and Q_{ex} the generalized external forces.

In order to solve the dynamic equations by numerical integration, one has to constitute the DAE system with the general form

$$\begin{cases} \ddot{\Phi}(q,t) = 0, \\ m\ddot{q} + J^T \lambda = Q_{ex}. \end{cases} \qquad (13)$$

For the sample mechanism, one could write 15 differential equations and 7·2=14 constraint equations, in total, a set of 29 equations with 29 unknowns: 15 generalized coordinates and 14 Lagrange multipliers.

4. The multi-particle system (MPS) model

This model considers the mechanism as a collection of particles subject to a set of absolute and relative constraints. Some principles of this model have been partially utilized in their work by Alexandru *et al*[1], Jalon - Bayo [4] and Geradin - Cardona [2].

The mechanism representation includes a *particle based model* for the rigid body and *point contact models* for each type of joint.

The *body model* consists in a set of particles separated by constant distances, each particle being associated with a concentrated mass according to the inertial equivalence with the real object. For a body model in 3D

space (i.e. able to integrally conserve the mass properties of the original solid) minimum 4 particles are needed and 3 particles for the planar case. Once the position of the particles is established in the body frame, the concentrated masses can be easily obtained from the inertial equivalence conditions.

In the 3D space a particle have 3 degrees of freedom (f = 3), therefore maximum three types of constraints can be imposed (figure 4):

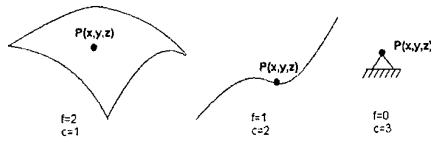


Figure 4

(i) Coincidence with a point (or another particle) → f = 0, c = 3.
(ii) Contact with a 3D curve → f = 1, c = 2.
(iii) Contact with a 3D surface → f = 2, c = 1.

The full body model includes the modelling particles and a set of constant distances constraints between them, which represent the ideal rigid conditions, according to the usual definition of the rigid body. For a body represented by 4 particles involving 4×3=12 generalized coordinates, a number of 6 distance constraints have to be imposed resulting finally only 6 independent coordinates (figure 5):

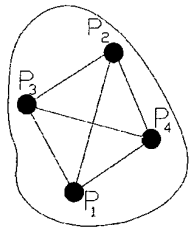


Figure 5

$$\begin{cases} (x_{P_1}-x_{P_2})^2+(y_{P_1}-y_{P_2})^2+(z_{P_1}-z_{P_2})^2 = P_1P_2^{\,2} \\ (x_{P_1}-x_{P_3})^2+(y_{P_1}-y_{P_3})^2+(z_{P_1}-z_{P_3})^2 = P_1P_3^{\,2} \\ (x_{P_2}-x_{P_3})^2+(y_{P_2}-y_{P_3})^2+(z_{P_2}-z_{P_3})^2 = P_2P_3^{\,2} \\ (x_{P_1}-x_{P_4})^2+(y_{P_1}-y_{P_4})^2+(z_{P_1}-z_{P_4})^2 = P_1P_4^{\,2} \\ (x_{P_2}-x_{P_4})^2+(y_{P_2}-y_{P_4})^2+(z_{P_2}-z_{P_4})^2 = P_2P_4^{\,2} \\ (x_{P_3}-x_{P_4})^2+(y_{P_3}-y_{P_4})^2+(z_{P_3}-z_{P_4})^2 = P_3P_4^{\,2} \end{cases} \qquad (14)$$

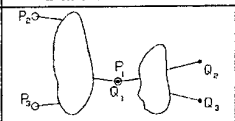
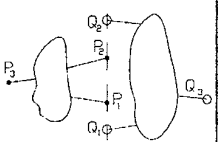
The *joint model* is defined as combination of constraints between the particles composing the two adjacent bodies. Each joint could be represented by a set of constraints defined between the particles of the two adjacent bodies. The point type contact model allows the definition of practically any type of joints. The models of the most usual joints are detailed in table I. With these models defined for body and joint, a new criterion can be formulated for the mechanism mobility as:

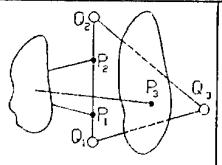
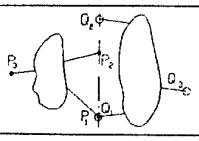
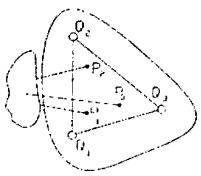
$$M = S \cdot p - \Sigma c_i, \qquad (15)$$

in which p is the number of the particles included in the model, S is the space dimension (S=3 for 3D space and S=2 for 2D space) and c_i is the number of constraints. The generalized coordinates vector has the form:

$$[q] = [x_1 \quad y_1 \quad z_1 \quad x_2 \quad y_2 \quad z_2 \quad x_3 \quad y_3 \quad z_3 \quad \ldots \quad x_p \quad y_p \quad z_p]^T, \qquad (16)$$

TABLE I. The usual joints representation for the multi-particle system (MPS) model

Joint type	Particle model	Constraints	Constraints Equation
Spherical joint		$P_1 \equiv Q_1$ c = 3 f = 3	$x_{P1}=x_{Q1}$, $y_{P1}=y_{Q1}$, $z_{P1}=z_{Q1}$.
Cylindrical joint		$P_1 \in Q_1Q_2$ axis $P_2 \in Q_1Q_2$ axis c = 4 f = 2	$\frac{x_{P_1}-x_{Q_1}}{x_{Q_2}-x_{Q_1}}=\frac{y_{P_1}-y_{Q_1}}{y_{Q_2}-y_{Q_1}}=\frac{z_{P_1}-z_{Q_1}}{z_{Q_2}-z_{Q_1}}$ $\frac{x_{P_2}-x_{Q_1}}{x_{Q_2}-x_{Q_1}}=\frac{y_{P_2}-y_{Q_1}}{y_{Q_2}-y_{Q_1}}=\frac{z_{P_2}-z_{Q_1}}{z_{Q_2}-z_{Q_1}}$

Translation joint		$P_1 \in Q_1Q_2$ axis $P_2 \in Q_1Q_2$ axis $P_3 \in Q_1Q_2Q_3$ plane c = 5 f = 1	Idem and $\begin{vmatrix} x_{P_3} & y_{P_3} & z_{P_3} & 1 \\ x_{Q_1} & y_{Q_1} & z_{Q_1} & 1 \\ x_{Q_2} & y_{Q_2} & z_{Q_2} & 1 \\ x_{Q_3} & y_{Q_3} & z_{Q_3} & 1 \end{vmatrix} = 0.$
Revolute joint		$P_1 \equiv Q_1$ $P_2 \in Q_1Q_2$ axis c = 5 f = 1	$x_{P1}=x_{Q1}, y_{P1}=y_{Q1}, z_{P1}=z_{Q1},$ $\frac{x_{P_2}-x_{Q_1}}{x_{Q_2}-x_{Q_1}} = \frac{y_{P_2}-y_{Q_1}}{y_{Q_2}-y_{Q_1}} = \frac{z_{P_2}-z_{Q_1}}{z_{Q_2}-z_{Q_1}}$
Plane joint		$P_1 \in Q_1Q_2Q_3$ plane $P_2 \in Q_1Q_2Q_3$ plane $P_3 \in Q_1Q_2Q_3$ plane c = 3 f = 3	$\begin{vmatrix} x_{P_i} & y_{P_i} & z_{P_i} & 1 \\ x_{Q_1} & y_{Q_1} & z_{Q_1} & 1 \\ x_{Q_2} & y_{Q_2} & z_{Q_2} & 1 \\ x_{Q_3} & y_{Q_3} & z_{Q_3} & 1 \end{vmatrix} = 0,$ i=1,2,3.

The vector [q] can be obtained by numeric solving of the system of M + Σc_i algebraic equations corresponding to the M driving motions and Σc_i joint constraints. It must be noted that for the a MPS model the joint equations can take only four possible forms, as resulting also from the table I:

- Distance equation

$$(x_{P_1}-x_{P_2})^2+(y_{P_1}-y_{P_2})^2+(z_{P_1}-z_{P_2})^2 = P_1P_2^2 \qquad (17)$$

- Coincidence equation

$$x_{PI}=x_{QI}, y_{PI}=y_{QI}, z_{PI}=z_{QI}, \qquad (18)$$

- Co-linearity equations

$$\frac{x_{P_1}-x_{Q_1}}{x_{Q_2}-x_{Q_1}} = \frac{y_{P_1}-y_{Q_1}}{y_{Q_2}-y_{Q_1}} = \frac{z_{P_1}-z_{Q_1}}{z_{Q_2}-z_{Q_1}} \qquad (19)$$

- Co-planarity equation

$$\begin{vmatrix} x_{P_3} & y_{P_3} & z_{P_3} & 1 \\ x_{Q_1} & y_{Q_1} & z_{Q_1} & 1 \\ x_{Q_2} & y_{Q_2} & z_{Q_2} & 1 \\ x_{Q_3} & y_{Q_3} & z_{Q_3} & 1 \end{vmatrix} = 0. \qquad (20)$$

In the next step, through successive differentiation, velocity and acceleration equations can be easily derived resulting the set of kinematic equations of the general form (5).

For dynamic simulation, the equations have the same general form as for MBS model - relation (12), in which the mass matrix is

$$m = diag[\ m_1\ \ m_1\ \ m_1\ \ m_2\ \ m_2\ \ m_2\ \ m_3\ \ m_3\ \ m_{31}\ \ \dots\ \ m_p\ \ m_p\ \ m_p\ ]. \qquad (21)$$

The Σci Lagrange multipliers include the joint reaction forces (including no torques) and also the internal cohesion forces between the particles of the bodies.

For the sample mechanism modelled as in figure 6, the number of particles per body is 2, except bodies 3 and 5, which are defined with three particles each. The total number of mobile particles is p = 12 (A_1, B_1, B_2, C_2, C_3, D_3, E_3, E_4, F_4, F_5, G_5, H_5), that is S·p=2×12=24 generalized coordinates (two Cartesian coordinates for each particle):

$$\boldsymbol{q} = [x_{A1}\ \ y_{A1}\ \ z_{A1}\ \ x_{B1}\ \ y_{B1}\ \ z_{B1}\ \ x_{B2}\ \ y_{B2}\ \ z_{B2}\ \ \dots\ \ x_{H5}\ \ y_{H5}\ \ z_{H5}]^T, \qquad (22)$$

As constraints, there are 9 rigid body constant distances (AB, BC, CD, DE, CE, EF, FG, FH, GH) and 14 joint constraints, yielding Σci=23, that is M=S·p-Σc_i=24-23=1. The constraint equations set is:

$$\begin{cases} (x_{A_1}-x_{B_1})^2+(y_{A_1}-y_{B_1})^2=l_1^2 \\ (x_{B_2}-x_{C_2})^2+(y_{B_2}-y_{C_2})^2=l_2^2 \\ (x_{C_3}-x_{D_3})^2+(y_{C_3}-y_{D_3})^2=l_3^2 \\ (x_{C_3}-x_{E_3})^2+(y_{C_3}-y_{E_3})^2=l_{CE}^2 \\ (x_{D_3}-x_{E_3})^2+(y_{D_3}-y_{E_3})^2=l_{DE}^2 \\ (x_{E_4}-x_{F_4})^2+(y_{E_4}-y_{F_4})^2=l_4^2 \\ (x_{F_5}-x_{G_5})^2+(y_{F_5}-y_{G_5})^2=l_{FG}^2 \\ (x_{F_5}-x_{H_5})^2+(y_{F_5}-y_{H_5})^2=l_{FH}^2 \\ (x_{G_5}-x_{H_5})^2+(y_{G_5}-y_{H_5})^2=l_{GH}^2 \end{cases}, \quad \begin{cases} x_{A_1}-x_{A_0}=0, & y_{A_1}-y_{A_0}=0 \\ x_{B_2}-x_{B_1}=0, & y_{B_2}-y_{B_1}=0 \\ x_{C_3}-x_{C_2}=0, & y_{C_3}-y_{C_2}=0 \\ x_{D_3}-x_{D_0}=0, & y_{D_3}-y_{D_0}=0 \\ x_{E_4}-x_{E_3}=0, & y_{E_4}-y_{E_3}=0 \\ x_{F_5}-x_{F_4}=0, & y_{F_5}-y_{F_4}=0 \\ y_{H_5}=a, & y_{G_5}=a \end{cases} \quad (23)$$

The 24th equation corresponds to the driving motion.

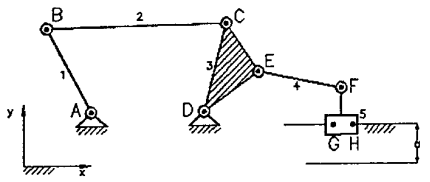


Figure 6.

The velocity and acceleration equations are derived by differentiation of the position equations

$$J\dot{q}=\Omega$$

$$J\ddot{q}=\psi\ ,$$

where the 24×24 Jacobian matrix J has also a very simple expression not given here, for space reasons.

For the dynamic analysis, one has to take into account the particles are acted by external, reaction and inertia forces. Each force is applied to a particle such as no torque is involved, which is an important simplification. The general matrix form of the differential equations is given also by (12), in which the mass matrix is 24×24

$m = diag[\ m_{A_1}\ m_{A_1}\ m_{A_1}\ m_{B_1}\ m_{B_1}\ m_{B_1}\ m_{B_2}\ m_{B_2}\ m_{B_2}\ \ldots\ m_{H_5}\ m_{H_5}\ m_{H_5}\ ]$ the 23×24 Jacobian matrix corresponds to the joint constraints and constant distances and the Lagrange multiplier vector λ has also 23 components. The DAE system has 47 equations with 47 unknown: 24 generalized coordinates and 23 Lagrange multipliers.

5. Conclusions

The model based approach for multibody simulation allows a clear background for any formulation including specific formulas for mobility computation and finally a proper evaluation of the potential for each of them. In spite of the larger number of equations, the MPS model provides several features with relevance to the non-linear multibody simulation:

- The representation of forces and inertial mass properties is significantly simplified.
- The constraints and the corresponding algebraic equations are of small variety -only four types of equations: distance equations, coincidence equations, co-linearity equations and co-planarity equations. This is simplifying both constraint and Jacobian matrix formulation.
- The MPS model allows the extension for the treatment of flexible multibody systems by introducing in the RHS of the distance equations the flexibility principles for each body as shown in [2].

6. References

1. Alexandru, P., Vişa, I., Talaba, D. (1988) Utilisation of the Cartesian coordinates for the linkages study (in Romanian). *The Romanian Symposium on Mechanisms and Machine Theory MTM '88*, Politehnic Institute of Cluj-Napoca, vol. I, 1-10.
2. Geradin, M. and Cardona, A. (2001) *Flexible Multibody Dynamics*, John Wiley & Sons, New York.
3. Haug, J.E. (1989) *Computers Aided Kinematics and Dynamics of Mechanical System*, vol. I. Ed. Allyn and Bacon.
4. Jalon, J.G. and Bayo, E. (1994) *Kinematic and Dynamic Simulation of Multibody Systems*, Springer-Verlag, New York.
5. Orlandea, N. Chace, M.A. and Calahan, D.A. (1977) A Sparsity Oriented Approach to the Dynamics Analysis and Design of Mechanical Systems, *Journal of Engineering for Industry*, August, 773-784.
6. Roberson, R. and Schwertassek, R. (1988) *Dynamics of Multibody Systems*. Springer Verlag, Berlin-New York.
7. Schiehlen, W.O. (1990) *Multibody Systems Handbook*, Springer Verlag, Berlin-New York.
8. Wittemburg, J. (1984) Analytical Methods in the Dynamics of Multibody Systems. *Proc. IUTAM / ISIMM Symposium on Modern Developments in Analytical Mechanics*, Turin, 835-858.

MBS Time Integration -Projective Constraint Violation Stabilization Methods on Manifolds

Zdravko Terze
University of Zagreb
Faculty of Mech. Eng. and Naval Arch.
Department of Aerospace Engineering
Ivana Lučića 5, 10000 Zagreb, Croatia

Dirk Lefeber
University of Brussels
Faculty of Applied Sciences
Department of Mechanical Engineering
Pleinlaan 2, B-1050 Brussels, Belgium

1. Unconstrained MBS on manifolds

Unconstrained multibody system (MBS) is an autonomous Lagrangian system. If n DOF is assumed, the system evolution in configuration space R^n is described (by definition) by Lagrangian equations [1]

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{\mathbf{x}}}\right)-\frac{\partial L}{\partial \mathbf{x}}=\Gamma^* \ , \ \mathbf{M}(\mathbf{x})\ddot{\mathbf{x}}=\mathbf{Q}^*(\mathbf{x},\dot{\mathbf{x}},t)\ , \tag{1}$$

By taking differentiable manifold approach, the configuration space R^n is considered to be a manifold M^n covered by coordinate system $\mathbf{x}(t)$ (in mathematical jargon of modern differential geometry: locally covered by chart $\mathbf{x}$). The solution of (1) is a dynamical trajectory $\mathrm{T}: x^i = x^i(t)$ of the system in n-dimensional manifold of configuration M. With every point on manifold of configuration, $\mathbf{x}\in\mathsf{M}$, the n dimensional tangent space $T_{\mathbf{x}}\mathsf{M}$ is affiliated, where system virtual displacements $\delta\mathbf{x}$ and velocities $\dot{\mathbf{x}}$ are contained, $\delta\mathbf{x}\in T_{\mathbf{x}}\mathsf{M}$, $\mathrm{d}\mathbf{x}\in T_{\mathbf{x}}\mathsf{M}$, $\dot{\mathbf{x}}\in T_{\mathbf{x}}\mathsf{M}$. The manifold M and the union of all tangent spaces at the various points $\mathbf{x}$ make another, $2n$ dimensional, manifold called tangent bundle, $T\mathsf{M}:\bigcup_{\mathbf{x}\in\mathsf{M}^n} T_{\mathbf{x}}\mathsf{M}$, covered by the coordinates $\mathbf{x},\dot{\mathbf{x}}: T\mathsf{M}=\{(\mathbf{x},\dot{\mathbf{x}}):\mathbf{x}\in\mathsf{M},\dot{\mathbf{x}}\in T_{\mathbf{x}}\mathsf{M}\}$ [2] (being mathematically not very rigorous, tangent bundle can be observed as a velocity phase space known from 'traditional' approach). Manifold M is not a vector space. By adopting system generalized mass matrix $\mathbf{M}(\mathbf{x})$ (positive definite) as a Riemannian metric on the manifold of configuration [3], a scalar product in the each tangent space $T_{\mathbf{x}}\mathsf{M}$ is given by $\langle\mathbf{y},\mathbf{z}\rangle_{\mathbf{M}(\mathbf{x})}=\mathbf{y}^{\mathrm{T}}\mathbf{M}(\mathbf{x})\mathbf{z}$, $\mathbf{y},\mathbf{z}\in T_{\mathbf{x}}\mathsf{M}$ [4]. Now, with the metric so defined, the tangent space $T_{\mathbf{x}}\mathsf{M}$ ('the fiber of the tangent bundle at point $\mathbf{x}$') becomes a local Euclidean vector space spanned by covariant basis $\hat{\mathbf{g}}_{\mathbf{x}_i}$. By introducing a reciprocal contravariant basis $\hat{\mathbf{g}}_{\mathbf{x}}^i$, the vectors in tangent spaces can be expressed using their contravariant and covariant representations $\hat{\dot{\mathbf{x}}}=\dot{x}^i\hat{\mathbf{g}}_{\mathbf{x}_i}$, $\dot{\mathbf{x}}=[\dot{x}^i]$, $\hat{\dot{\mathbf{x}}}=\dot{x}_i\hat{\mathbf{g}}_{\mathbf{x}}^i$, $\dot{\mathbf{x}}^*=[\dot{x}_i]$.

2. Geometric properties of constraints

2.1 Holonomic constraints

Holonomic constraints

$$\Phi(\mathbf{x},t)=\mathbf{0}\ , \quad \Phi(\mathbf{x},t):\mathsf{R}^n\times\mathsf{R}\rightarrow\mathsf{R}^r\ , \tag{2}$$

that are imposed on the system

a) restrict system configuration space ('positions'): a trajectory $\mathrm{T}: x^i = x^i(t)$ 'moves' on the n-r dimensional constraint manifold $\mathsf{S}^{n-r}(t)$, $\mathsf{S}^{n-r}(t)=\{\mathbf{x}\in\mathsf{M},\Phi(\mathbf{x},t)=\mathbf{0}\}$, $t\geq 0$, $\mathbf{x}(t_0)\in\mathsf{S}^{n-r}(t_0)$,

b) at the velocity level they induce constraint equation

$$\Phi^*_{\mathbf{x}}(\mathbf{x},t)\dot{\mathbf{x}} = -\Phi_t = \tau \tag{3}$$

that is linear in velocities. The constraint matrix $\Phi^*_{\mathbf{x}}(\mathbf{x},t)$ can be written in the form $\Phi^{*\mathrm{T}}_{\mathbf{x}}(\mathbf{x},t) = [\varphi^*_1,....,\varphi^*_r]$, $\varphi^*_i = [\varphi_i]$, $\hat{\varphi}_1 = \varphi_i\, \hat{\mathbf{g}}^i_{\mathbf{x}}$. The vectors $\hat{\varphi}_1,...,\hat{\varphi}_r$ represent gradients to the constraint hypersurfaces. They are linearly independent and span r dimensional constraint subspace $\mathsf{C}^r_{\mathbf{x}}$ [6]. Kinematically admissible virtual displacements $\delta\mathbf{x}$ are restricted to the n-r dimensional tangent space $T_{\mathbf{x}}\mathsf{S}^{n-r}$ that is orthogonal to $\mathsf{C}^r_{\mathbf{x}}$. Together, subspaces $\mathsf{C}^r_{\mathbf{x}}$ and $T_{\mathbf{x}}\mathsf{S}^{n-r}$ span fiber of tangent bundle of unconstrained system $T_{\mathbf{x}}\mathsf{M}^n$ at point $\mathbf{x}$: $T_{\mathbf{x}}\mathsf{S}^{n-r} \cap \mathsf{C}^r_{\mathbf{x}} = 0$, $T_{\mathbf{x}}\mathsf{S}^{n-r} \cup \mathsf{C}^r_{\mathbf{x}} = T_{\mathbf{x}}\mathsf{M}^n$.

2.2 Non-holonomic constraints

If, beside h holonomic constraints (2), the additional nh non-holonomic constraints

$$\Psi(\mathbf{x},\dot{\mathbf{x}},t) = \mathbf{0} \tag{4}$$

are imposed on the system, they do not restrict system configuration space (system constraint manifold S^{n-r} maintains the same dimension, $r = h$) but impose additional velocity constraints on holonomic constraint manifold tangent bundle $T\mathsf{S}$, $\dot{\mathbf{x}} \in T^{n-r-nh}_{\mathbf{x}}\mathsf{S}^{n-r} \subset T^{n-r}_{\mathbf{x}}\mathsf{S}^{n-r}$.

If non-holonomic constraints are linear in velocities, i.e. can be given in Pfaffian form

$$\Psi = \mathbf{B}^*(\mathbf{x},t)\dot{\mathbf{x}} - \beta(\mathbf{x},t) = \mathbf{0}, \tag{5}$$

the system constraint equations can be written as follows:

$$\begin{bmatrix}\Phi^*_{\mathbf{x}}(\mathbf{x},t)\\ \mathbf{B}^*(\mathbf{x},t)\end{bmatrix}\dot{\mathbf{x}} = \begin{bmatrix}\tau\\ \beta\end{bmatrix} \quad , \quad \begin{bmatrix}\Phi^*_{\mathbf{x}}(\mathbf{x},t)\\ \mathbf{B}^*(\mathbf{x},t)\end{bmatrix} = \Phi^*_{\mathbf{x}\,nh} \quad , \quad \Phi^*_{\mathbf{x}\,nh} \in \mathsf{R}^{(h+nh)\times n}. \tag{6}$$

3. Projective constraint stabilization method via coordinates partitioning

If system governing equations are expressed in descriptor form, a constraint violation stabilization method have to be applied during integration procedure. A well known method that provides a full constraint stabilization is generalized coordinates partitioning procedure [5]. If the system is holonomic and constrained on manifold $\mathsf{S}^{n-r}(t)$, the 'classical' coordinate partitioning algorithm is based on pivoting operations on the constraint matrix $\Phi^*_{\mathbf{x}}$, $\mathrm{rank}(\Phi^*_{\mathbf{x}}) = r$, by means of which the subvectors of dependent and independent coordinates $\mathbf{x}^{\mathrm{d}} \in \mathsf{R}^r$ and $\mathbf{x}^{\mathrm{i}} \in \mathsf{R}^{n-r}$ is extracted.

With the attempt to provide a further insight into the characteristics of the method, in this paper, the coordintes partitioning algorithm will be analysed on manifolds using differential geometry approach.

Criteria for partitioning can be expressed geometrically: basically, every partitioning that returns subvector of dependent coordinates $\mathbf{x}^{\mathrm{d}}$ which basis vectors have non-zero projections on the constraint subspace $\mathsf{C}^r_{\mathbf{x}}$ (the corresponding $r\times r$ submatrix of constraint matrix $\Phi^*_{\mathbf{x}}$ is non-singular) is correct one and can be used for stabilization procedure. Consequently, the basis vectors of variables $\mathbf{x}^{\mathrm{i}}$ have projections on tangent space of constraint manifold $T_{\mathbf{x}}\mathsf{S}^{n-r}$ that is complement to $\mathsf{C}^r_{\mathbf{x}}$. If the extracted subvectors do not satisfy specified conditions, the partition is not a valid one and the calculation will fail.

After partitioning, time integration results of the system variables $\dot{\mathbf{x}}, \mathbf{x}$ are projected to the constraint manifold tangent bundle $T\mathbf{S}$ to assure full satisfaction of the system constraints. This can be achieved by correcting dependent coordinates $\mathbf{x}^d$ to bring the configuration coordinates $\mathbf{x}$ in accordance with the constraint equation (2) up to the required accuracy (a projection on $\mathbf{S}^{n-r}$ can be accomplished by iterative solving of (2), while keeping values of independent coordinates unchanged and treating $\mathbf{x}^d$ as unknown variables). The procedure is than repeated at the velocity level by correcting $\dot{\mathbf{x}}^d$ to bring $\dot{\mathbf{x}}$ in accordance with (3), with the only difference that (3) is linear algebraic system and $\dot{\mathbf{x}}^d$ can be obtained straightforwardly.

The main problem that may occur during stabilization procedure is an inadequate coordinate partitioning that can have a negative effect on the integration accuracy along constraint manifold [7]. Although, as it was explained, every partitioning that returns the acceptable subvectors can be used for the stabilization procedure providing the constraint stabilization, a non-optimal choice of the subvectors can cause an increase of the numerical errors along manifold during stabilization procedure (numerical errors along constraint manifold affect system evolution in time i.e. its kinetic motion). It means that, in this case, a correction of the constraint violation will be accomplished at the expense of the 'kinetic motion' accuracy obtained by the system variables $\dot{\mathbf{x}}, \mathbf{x}$ ODE integrators.

3.1. Stabilization of the system configuration constraints

The 'mechanism' of emerging of the numerical errors along configuration manifold, because of an inadequate partitioning during the stabilization procedure of holonomic systems, is outlined in Fig. 1, where an illustrative example $\mathbf{x} \in \mathbf{R}^2$, $\mathbf{S}^1$ is discussed.

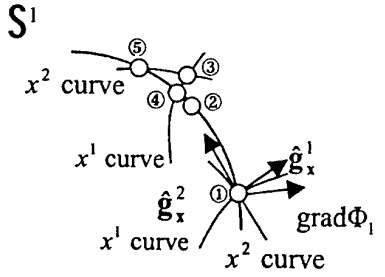


Fig. 1: Correction of the configuration constraint violation

Assuming that, starting from position ①, an integration of ODE gives result ③ instead of exact position ② (a scleronomic system is assumed), a projection on the constraint manifold $\mathbf{S}^1$ by adjusting coordinate x^1 (solving 'position' i.e. configuration constraint equation (2) along x^1curve by treating x^1 as dependent i.e. unknown variable) yields result ④ that is consistent to the constraint. If instead of x^1, the variable x^2 was chosen to be a dependent coordinate, an adjustment of the integration result along x^2 curve would yield solution ⑤, which is also consistent to the constraint but contains considerable error along the manifold $\mathbf{S}^1$.

A remedy for the problem of an inadequate partitioning has been offered in [8], where a projective criterion to the coordinate partitioning method is introduced (for application, see [9]). For a given set of coordinates of unconstrained system, the criterion allows for the optimal choice of dependent/independent coordinates which, consequently, gives opportunity to minimize integration error along manifold.

The main idea is to determine those r coordinates which direction vectors $\hat{\mathbf{g}}_{\mathbf{x}}^{i}$ deliver the biggest relative projections to the $\mathsf{C}_{\mathbf{x}}^{r}$ and select them as dependent variables which will be adjusted during the stabilization procedure. By correcting the coordinates whose direction vectors align well with the constraint gradients (that point directions toward constraint surfaces and span $\mathsf{C}_{\mathbf{x}}^{r}$), it is ensured that the correction procedure will shift a state-point of the system 'as direct as possible' to the constraint hypersurfaces, minimising thus an error along constraint manifold. Along this line, in the example shown in Fig. 1, the variable x^1 is chosen to be a dependent coordinate since its basis vector $\hat{\mathbf{g}}_{\mathbf{x}}^{1}$ delivers a big projection on $\mathsf{C}_{\mathbf{x}}^{r}=\mathrm{grad}[\Phi_1=0]$ (in this illustrative example the constraint subspace $\mathsf{C}_{\mathbf{x}}^{r}$ is one-dimensional, spanned by $\mathrm{grad}[\Phi_1=0]$).

3.2 Stabilization of the velocity constraints

The projective criterion to the coordinate partitioning method can be utilizied for a minimization of the numerical errors in the process of correction of constraint violation at the velocity level as well. Here, an application of the criterion enhances a definitness of the velocity constraints algebraic system (3), providing thus a better numerical accuracy of the stabilization procedure. This feature is illustrated by an example $\mathbf{x}\in\mathsf{R}^3$, $\mathsf{S}^2=\{\mathbf{x}\in\mathsf{R}^3,\Phi_1(\mathbf{x})=0\}$, shown in Fig. 2. For seak of simplicity, scleronomic system and orthogonal basis $\hat{\mathbf{g}}_{\mathbf{x}}^{1},\hat{\mathbf{g}}_{\mathbf{x}}^{2},\hat{\mathbf{g}}_{\mathbf{x}}^{3}$ are assumed. If the velocity equation (3) is written in the 'vectorial' form, for the analysed case it reads

$$\mathrm{grad}\,\Phi_1\cdot\dot{\hat{\mathbf{x}}}=0\quad . \tag{7}$$

In (7), the components of $\mathrm{grad}\,\Phi_1$ represent coefficients of the linear algebraic system that, for a general mathematical model, is given by (3). By applying the projective criterion and choosing $\dot{x}^3$, which direction vector $\hat{\mathbf{g}}_{\mathbf{x}}^{3}$ (in this 'academic' illustrative situation) is almost collinear to $\mathrm{grad}\,\Phi_1$, as a dependent coordinate, the potential numerical errors in independent coordinates $\dot{x}^1$ and $\dot{x}^2$ would not affect considerably the solution $\dot{x}^3\approx 0$ of the velocity constraint equation (7).
This is because of the small magnitudes of the coordinates of $\mathrm{grad}\,\Phi_1$ along the basis vectors $\hat{\mathbf{g}}_{\mathbf{x}}^{1}$ and $\hat{\mathbf{g}}_{\mathbf{x}}^{2}$ (small projections of $\mathrm{grad}\,\Phi_1$ on $\hat{\mathbf{g}}_{\mathbf{x}}^{1}$ and $\hat{\mathbf{g}}_{\mathbf{x}}^{2}$) that multiply $\dot{x}^1$ and $\dot{x}^2$ while solving (7) for $\dot{x}^3$.

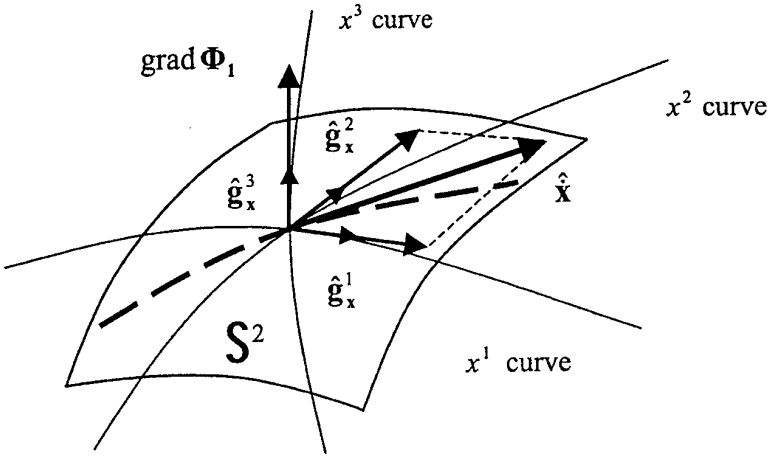


Figure 2: Correction of the constraint violation at the velocity level

4. Structure of the partitioned subvectors

To gain a further insight into the partitioning procedure and its characteristics, it is illustrative to observe the algorithm of the projective criterion at the tangent bundle $T\mathbf{M} = \{ (\mathbf{x}, \dot{\mathbf{x}}) : \mathbf{x} \in \mathbf{M},\ \dot{\mathbf{x}} \in T_{\mathbf{x}}\mathbf{M} \}$ of an unconstrained system. As explained, $T\mathbf{M}$ is $2n$-dimensional Riemannian manifold with a metric $\mathbf{M}_{TM} = \mathrm{diag}(\mathbf{M}(\mathbf{x}), \mathbf{M}(\mathbf{x}))$, where a configuration of the system as well as its velocities can be studied [10]. If constraints are present, they are represented in $T\mathbf{M}$ by the configuration and velocity submanifolds, by means of which the possible states of system are determined. Observed at $T\mathbf{M}$, the partitioning procedure for constraints stabilization can be studied for each submanifolds separately. By using the projective criterion for both manifolds, characteristics of the partitioning procedure that for a given set of coordinates $\mathbf{x} \in \mathbf{M},\ \dot{\mathbf{x}} \in T_{\mathbf{x}}\mathbf{M}$ provides the optimal dependent/independent subvectors, can be learned as follows.

4.1 Holonomic constraints

The configuration submanifold $\mathbf{S}^{n-r}$ is determined by the equation (2) i.e.

$$\mathbf{S}^{n-r} = \{ \mathbf{x} \in \mathbf{M}, \Phi(\mathbf{x}, t) = \mathbf{0} \} . \tag{8}$$

The submanifold $\mathbf{V}^{n-r}$, by means of which the system velocities $\dot{\mathbf{x}}$ are constrained, is defined by (3), thus

$$\mathbf{V}^{n-r} = \{ \dot{\mathbf{x}} \in T_{\mathbf{x}}\mathbf{M} ,\ \Phi^{*}{}_{\mathbf{x}}(\mathbf{x}, t)\, \dot{\mathbf{x}} = \boldsymbol{\tau} \} . \tag{9}$$

If the optimization projective criterion is applied during the partitioning procedure at the both configuration and velocity level (which is the most common procedure), the criterion itself is based on determination of the gradients to the constraint submanifolds $\mathbf{S}^{n-r}$ and $\mathbf{V}^{n-r}$ (as explained, this is because the extraction of the dependent coordinates of $\mathbf{x}^{\mathrm{d}}$ and $\dot{\mathbf{x}}^{\mathrm{d}}$ depend on the directions of gradients to the hypersurfaces of submanifolds $\mathbf{S}^{n-r}$ and $\mathbf{V}^{n-r}$ respectively).

Since constraint submanifold $\mathbf{S}^{n-r}$ is determined by (2), the $\mathbf{x}$ correction gradient by means of which $\mathbf{x}^{\mathrm{d}}$ is to be extracted is given by

$$\mathrm{grad}\,[\Phi(\mathbf{x}, t) = \mathbf{0}] = \Phi^{*}_{\mathbf{x}}(\mathbf{x}, t) . \tag{10}$$

Similarly, $\dot{\mathbf{x}}$ correction gradient, decisive for an extraction of $\dot{\mathbf{x}}^{\mathrm{d}}$ reads as

$$\mathrm{grad}\,[\Phi^{*}{}_{\mathbf{x}}(\mathbf{x}, t)\, \dot{\mathbf{x}} = \boldsymbol{\tau}] = \Phi^{*}_{\mathbf{x}}(\mathbf{x}, t) . \tag{11}$$

Now, if the expressions (10) and (11) are compared, it is obvious that the both hypersurfaces $\mathbf{S}^{n-r}$ and $\mathbf{V}^{n-r}$ have the same gradients for every point in $T\mathbf{M}$ (in fact, the both gradients depend on the current position $\mathbf{x} \in \mathbf{M}$ at the configuration manifold and t only, i.e. they are independent on system velocities $\dot{\mathbf{x}}$). Of course, this stems from the fact that, in the case of holonomic systems, the velocity submanifold $\mathbf{V}^{n-r}$ is determined by algebraic equations (3) (linear in $\dot{\mathbf{x}}$!) which are, in turn, obtained by derivation of the configuration constraints (2).

Since the gradients to the both hypersurfaces $\mathbf{S}^{n-r}$ and $\mathbf{V}^{n-r}$ are identical, it is clear that, for holonomic systems, the optimal coordinate partitioning procedure provides the same optimal dependent/independent subvectors at the both configuration and velocity level i.e. $[x_1, x_2, \ldots, x_{\mathrm{d}}]^{\mathrm{T}}$ and $[\dot{x}_1, \dot{x}_2, \ldots, \dot{x}_{\mathrm{d}}]^{\mathrm{T}}$. This means that, once the partitioning procedure is accomplished for the configuration coordinates and subvector $\mathbf{x}^{\mathrm{d}}$ is extracted, it is not needed to be repeated at the velocity level (the subvector $\dot{\mathbf{x}}^{\mathrm{d}}$ of the same structure is to be chosen for the stabilization of velocities).

4.2 Non-holonomic constraints

A coordinates partitioning procedure can also be applied for stabilization of constraint violation of non-holonomic systems. If additional *nh* non-holonomic constraints (4), which are imposed on the system (beside *h* holonomic constraints (2) that define configuration manifold S^{n-r}, $r = h$), are given in linear (Pfaffian form) (5), the submanifold V^{n-r-nh} of the velocity constraints are defined by

$$\begin{bmatrix} \Phi_{\mathbf{x}}^{*}(\mathbf{x},t) \\ \mathbf{B}^{*}(\mathbf{x},t) \end{bmatrix} \dot{\mathbf{x}} = \Phi_{nh}^{*} \dot{\mathbf{x}} = \begin{bmatrix} \tau \\ \beta \end{bmatrix} \quad , \quad \Phi_{\mathbf{x}\,nh}^{*} \in R^{(h+nh)\times n} \quad . \tag{12}$$

By considering (12), the $\dot{\mathbf{x}}$ correction gradient reads as

$$\operatorname{grad}\left[\Phi_{nh}^{*} \dot{\mathbf{x}} = \begin{bmatrix} \tau \\ \beta \end{bmatrix} \right] = \Phi_{nh}^{*}(\mathbf{x},t) = \begin{bmatrix} \Phi_{\mathbf{x}}^{*}(\mathbf{x},t) \\ \mathbf{B}^{*}(\mathbf{x},t) \end{bmatrix} . \tag{13}$$

Since non-holonomic constraints do not affect configuration manifold S^{n-r}, the 'position' coordinates correction gradient is given by (10).

By comparing correction gradients (10) and (13) which do not match any more, it can be concluded that in the case of non-holonomic systems the optimal coordinates partitioning will not 'return' dependent/independent subvectors of the same structure for configuration and velocity stabilization. Beside non-equality of dimension of the subvectors $\mathbf{x}^{d} \in R^{r}$ and $\dot{\mathbf{x}}^{d} \in R^{r-nh}$, their structure will also differ in general case. Generally, in the case of non-holonomic systems, a separate partitioning procedure is necessary for stabilization at configuration and velocity level.

This is specially true if the imposed non-holonomic constraints (4) can not be put in Pfaffian form. If non-holonomic constraints are non-linear in velocities (this kind of constraints can appear as a result of certain controlling actions), it will be necessary to determine a completely new correction gradient

$$\operatorname{grad}\left[\Psi(\mathbf{x},\dot{\mathbf{x}},t) = \mathbf{0}\right] = \Psi_{\dot{\mathbf{x}}} \quad , \tag{14}$$

to accomplish optimal correction of the velocity constraint violation.

References

[1] V. I. Arnold, *Mathematical Methods of Classical Mechanics*, Springer-Verlag, New York (1978).

[2] R. Abraham, J. E. Marsden, T. Ratiu, *Manifolds, Tensor Analysis, and Applications*, Springer-Verlag, New York (1988).

[3] W. Blajer, A Geometric Unification of Constrained System Dynamics, *Multibody System Dynamics*, **1**, 3-21 (1997).

[4] U. Jungnickel, Differential-Algebraic Equations in Riemannian Spaces and Applications to Multibody System Dynamics, *Zeitschrift fuer Angewandte Mathematik und Mechanik*, **74**, 409-415 (1994).

[5] P. E. Nikravesh, *Computer-Aided Analysis of Mechanical Systems*, Prentice Hall (1988).

[6] W. Schiehlen, *Multibody Systems Handbook*, Springer-Verlag, Berlin (1990).

[7] Z. Terze, D Lefeber, Dynamic Simulation of Multibody Systems With no Constraints Violation, *Transactions of FAMENA*, **24**, 1-9 (2000).

[8] W. Blajer, W. Schiehlen and W. Schirm, A Projective Criterion to the Coordinate Partitioning Method for Multibody Dynamics, *Archive of Applied Mechanics*, **64**, 86-98 (1994).

[9] Z. Terze, D. Lefeber, O. Muftić, Null Space Integration Method for Constrained Multibody System Simulation with no Constraint Violation, *Multibody System Dynamics*, **6**, 229-243 (2001).

[10] J. V. Josė, E. J. Saletan, *Classical Dynamics – A Contemporary Approach*, Cambridge University Press, Cambridge, UK (1998).

Model Reduction Techniques for Vehicle Suspensions in Real-Time Applications

Jakub Tobolář (jakub.tobolar@dlr.de)
DLR German Aerospace Center, Vehicle System Dynamics Group,
P.O.Box 1116, D - 82230 Weßling, Germany

Abstract. Hardware-in-the-Loop test facilities become a modern tool for testing of active systems in the automotive industry. Since real-time models are necessary for the vehicle dynamic simulation in HiL various techniques have been developed in the recent years to fasten the time integration of multibody system models.

Nowadays a number of simplifications is well established that help to avoid the time consuming simulation of the suspensions with kinematic closed loops. The so called virtual axle is one of these approaches. It is based on tabulated kinematics of a wheel carrier that is evaluated during time integration. The virtual axle can be optionally extended with simplified wheel carrier elastokinematics as well.

This paper presents the implementation of the virtual axle method in the multibody simulation package SIMPACK. Furthermore the accuracy of a vehicle model with the virtual axle is compared with a classical model including full suspensions with kinematic closed loops.

Keywords: real-time simulation, suspension, virtual axle, model reduction

Abbreviations: HiL - Hardware-in-the-Loop; MBS - multibody systems

1. Introduction

To decrease development time and costs in automotive industry the multibody dynamic simulation programs are widely used. In many cases the number of ride tests can be decreased significaltly by using simulation models.

Since the multibody systems (MBS) simulation of vehicles has become a standard simulation tool the essential effort is payed in the recent years to decrease the computing time needed for the numerical integration. Especially in case of Hardware-in-the-Loop (HiL) tests the computing time is crucial.

Various MBS-formalisms and simulation techniques have been suggested to reduce the computing time in dynamic simulation, see [1–4]. The approaches deal with various aspects that can be summarised as follows

- techniques simplifying the model complexity,
- descriptor or state-space form of equations of motion, the type of coordinates,
- selected numerical integration methods and
- parallelisation techniques and used hardware.

The present paper is focused on the simplification of complex suspension systems.

1.1. SIMPLIFICATION OF SUSPENSION MODELS

The necessity of real-time MBS vehicle models (e.g. for HiL) is the motivation to perform significant model simplifications including methods to avoid the kinematic closed loops caused by suspension systems.

The resulting suspension model should be simple and precise at the same time. Nevertheless the accuracy is often less important in order to decrease the computing time of the real-time model.

The kinematic loops are caused by suspension design ensuring optimal behaviour of the suspension for the full range of wheel movement. Several approaches are used to simplify the full suspension model that generally leads to the solution of differential algebraic equations (DAE). The most often used forms are the following:

- Transformation of DAE to ordinary differential equations (ODE). This method can lead to numerical problems, see [5].
- Precalculation of the wheel carrier movement relative to the vehicle chassis and saving of data in table form to avoid the algebraic equations caused by kinematic loops, see Section 2.
- Solution of algebraic equations within the wheel carrier joint. Then the resulting equations of the complete vehicle form an ODE (Suspension Composite Joint description [6]).

Elastokinematics is a consequence of elastic bushings and compliance of bodies. It has a strong influence on the suspension movement if dynamical forces are considered that act on the suspension.

The elastokinematic model of suspension requires the iterative solution of a system of non-linear equations to get the resulting position since suspension position and compliance depend on each other. Generally the resulting position and orientation $\mathbf{p}_i$ of the i–th wheel carrier is

$$\mathbf{p}_i = \mathbf{p}_i(\mathbf{z}_i, \mathbf{Q}_i, t) \ , \tag{1}$$

where $\mathbf{z}_i$ are independent coordinates and $\mathbf{Q}_i$ are the acting forces.

For the real-time models the complex elastokinematics must be strongly simplified and for a lot of ride tests it can be even completely neglected. The dependency of compliance on the initial position without feedback is often considered to fasten the computation. Therefore within one time step (1) simplifies to

$$\mathbf{p}_{i,t_k} = \mathbf{p}_i(\mathbf{z}_{i,t_k}, t_k) + \Delta\mathbf{p}_i(\mathbf{z}_{i,t_k}, \mathbf{Q}_{i,t_{k-1}}, t_k). \tag{2}$$

The additive term $\Delta\mathbf{p}_i$ indicates simplified elastokinematics and it is a non-linear function of forces acting on the wheel carrier. The forces $\mathbf{Q}_{i,t_{k-1}}$ from previous time step t_{k-1} are selected since the actual forces of k-th step are not known during solution of equation (2). The iterative solution is avoided in this way.

The additive term is often linearised to the form (see [8])

$$\Delta\mathbf{p}_i = c_{e1}(\mathbf{z}_i)Q_{1i} + c_{e2}(\mathbf{z}_i)Q_{2i} + \cdots + c_{e6}(\mathbf{z}_i)Q_{6i}, \tag{3}$$

with compliance coefficients c_{ej}, $j = 1, \ldots, 6$ and a vector $\mathbf{Q}_i$ including three force Q_{ji}, $j = 1, \ldots, 3$ and three torque Q_{ji}, $j = 4, \ldots, 6$ components.

2. Virtual axle

The virtual axle is characterised by precalculated and tabulated data that is evaluated later during the dynamic simulation, see [7], [8]. The presented virtual axle has been implemented in the multibody simulation package SIMPACK, see [4]. In this simplified axle model the position and orientation $\mathbf{p}_i$ of the wheel carrier are calculated depending on a set of dependent coordinates $\mathbf{q}_i = [q_{1i} \cdots q_{6i}]^T$

$$\mathbf{p}_i = \mathbf{p}_i(\mathbf{q}_i). \tag{4}$$

The dependent coordinates $\mathbf{q}_i$ itself are generally defined by two independent coordinates $\mathbf{z}_i$ that are given by:

z_{1i} ... vertical movement of the wheel carrier r_{zi} and

z_{2i} ... rack rod displacement r_{yR} or steering angle δ.

The coordinates $\mathbf{q}_i$ are tabulated based on the data that is obtained from a MBS model of full suspension that defines the transformation

$$\mathbf{q}_i = \mathbf{q}_i(\mathbf{z}_i). \tag{5}$$

During time integration the data is evaluated with cubic tensor product splines that interpolate data tables. A lower order of interpolating spline could be selected but generally this leads to discontinuities and problems during numerical integration.

2.1. Kinematics

The position and orientation $\mathbf{p}_{ki}$, velocity $\mathbf{v}_{ki}$ and acceleration $\mathbf{a}_{ki}$ of wheel carrier relative to vehicle chassis are calculated in the virtual axle

$$\mathbf{p}_{ki} = \mathbf{p}_{ki}(\mathbf{z}_i, t) \tag{6}$$

$$\mathbf{v}_{ki} = \mathbf{v}_{ki}(\mathbf{z}_i, \dot{\mathbf{z}}_i, t) = \mathbf{J}_i(\mathbf{z}_i, t)\dot{\mathbf{z}}_i + \overline{\mathbf{v}}_i(\mathbf{z}_i, t) \tag{7}$$

$$\mathbf{a}_{ki} = \mathbf{a}_{ki}(\mathbf{z}_i, \dot{\mathbf{z}}_i, \ddot{\mathbf{z}}_i, t) = \mathbf{J}_i(\mathbf{z}_i, t)\ddot{\mathbf{z}}_i + \overline{\mathbf{a}}_i(\mathbf{z}_i, \dot{\mathbf{z}}_i, t) \tag{8}$$

where the subscript k denotes the (initial) kinematic solution.

The terms $\mathbf{J}_i$, $\overline{\mathbf{v}}_i$ and $\overline{\mathbf{a}}_i$ in equations (7) and (8) include the partial derivatives of transformation (5).

2.2. Elastokinematics

The resulting position can be influenced optionally by the elastokinematic term $\Delta\mathbf{p}_i$

$$\mathbf{p}_i = \mathbf{p}_{ki}(\mathbf{z}, t) + \Delta\mathbf{p}_i(\mathbf{z}_i, \mathbf{Q}_{i,t_{k-1}}, t). \tag{9}$$

with the joint constraint forces $\mathbf{Q}_{i,t_{k-1}}$ from the previous time step.

Both tabulated non-linear and linearised $\Delta\mathbf{p}_i$ may be selected. The later needs the tabulated coefficients c_{ej}, $j = 1,\ldots,6$, see Equation (3). The non-linear approach is more suitable to have a more exact model. Unfortunately it is rather difficult to get the tabulated input data. Therefore the linear method is used in general. Although additional error terms are introduced by superimposing particular linear terms, the simulation results are often of good approximation.

Just the quasi-static forces are realised by the precalculation of elastokinematics data since it is impossible to include the whole range of dynamical forces.

3. Simulation

In this section the vehicle model using the kinematic solution of virtual axle in SIMPACK is described and results of a ride manoeuvre are presented.

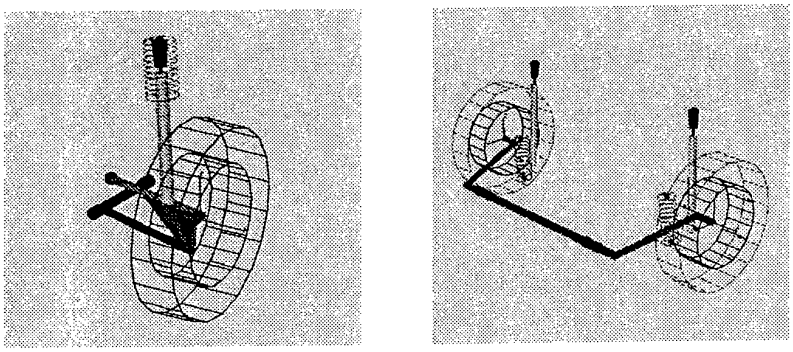
3.1. Vehicle model and ride manoeuvre

The model of a middle class vehicle has been chosen to compare the accuracy of the virtual axle with the complete suspension model. The vehicle has a McPherson front suspension (Figure 1.a) and a twist-beam rear axle (Figure 1.b). These are modelled as a complete suspension in the vehicle *Model 1.* The front suspension is substituted by the virtual axle in a simplified vehicle model (*Model 2*).

No steering system is considered but the simple time excitation of the rack rod displacement z_{2i}. The excitation $z_{2i} = z_{2i}(t)$ is applied directly to the virtual axle in the simplified *Model 2.*

Both the front and rear anti-roll-bars are treated as torsional springs/dampers. The brake assembly and power train are neglected because they are of lower importance for the performed manoeuvre.

The curve entry has been selected for comparison of the models. The initial velocity of vehicle is 10 m/s. The displacement of the rack rod (excitation z_{2i}) from 0.0 m to 0.033 m is defined in the time interval between 0.5 and 1.0 s. The total time of manoeuvre is 7 seconds.



a. Front suspension b. Rear suspension

Figure 1. Vehicle suspensions of the simulated vehicle.

3.2. RESULTS

Four vehicle models have been compared. The first one is *Model 1* the others are three variants of *Model 2* that differ by the selected interpolation:

- *Model 2a*: Cubic spline interpolation of $\mathbf{q}_i$ in direction z_{1i} and linear interpolation in direction z_{2i}.
- *Model 2b*: Cubic spline interpolation of $\mathbf{q}_i$ in both directions. All partial derivatives are neglected.
- *Model 2c*: Cubic spline interpolation of $\mathbf{q}_i$ in both directions. All partial derivatives are considered.

The SIMPACK integrator SODASRT with variable time step is used in all computations. The computations have been performed for three different integration tolerances. The computing times of all models are shown in Table I.

Figure 2 presents the vehicle position during the ride manoeuvre. The divergence of the results is especially obvious at the end of simulation. To point out the differences the selected area of Figure 2.a is focused in Figure 2.b again.

As can be seen in Figure 2 the results of *Model 1* and *Model 2c* are nearly the same. The *Model 2b* is faster at approximately 15% but it deviates at some 0.5 m as compared to *Model 2c*. The divergence of *Model 2a* can be seen as well.

In contrast to the divergence of *Model 2a* that increases with time the deviation of *Model 2b* changes slightly when the rack rod displacement stays constant and it arises mainly during the rack rod movement.

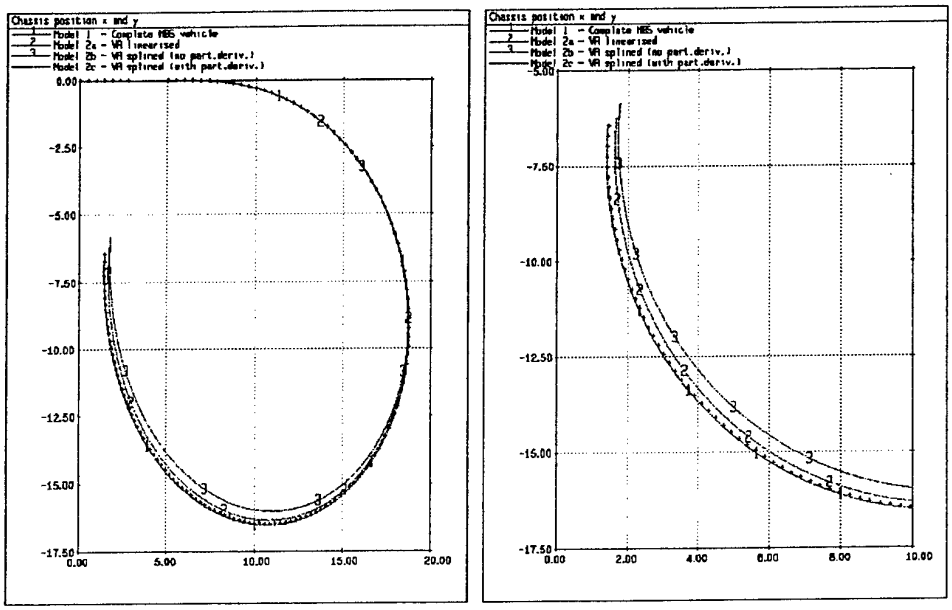
As can be seen in Table I it was not possible to calculate *Model 2a* with small tolerances because there have been discontinuities in partial derivatives of the selected linear interpolation.

4. Conclusions

In this study the virtual axle has been implemented in the SIMPACK simulation package. The reduced vehicle model with virtual axle has been compared with the full model. The results indicate that the virtual axle with cubic spline interpolation in both directions and partial derivatives is the most suitable solution to supply the complex suspension.

Table I. Computing times of simulated vehicle models [s].

Vehicle model	Tolerance 1.10^{-3}	1.10^{-4}	1.10^{-5}
1	3.57	4.82	6.8
2a	2.51	aborted	aborted
2b	2.28	3.78	5.24
2c	3.05	4.33	6.13



a. b.

Figure 2. Position of the vehicle in a horizontal plane.

The linear interpolation can be used, too, if no discontinuities occur during simulation or if the model accuracy is of lower importance.

References

1. Rill, G.: Vehicle Modeling for Real Time Applications, *RBCM - Journal of the Braz. Soc. Mechanical Sciences*, Vol. XIX, No.2, pp. 192–206, 1997.
2. Eichberger, A.: *Simulation von Mehrkörpersystemen auf parallelen Rechnerarchitekturen*, Fortschritt-Berichte VDI Reihe 8, in German, 1993.
3. Sailer, U.: *Nutzfahrzeug-Echtzeitsimulation auf Parallelrechnern mit Hardware-in-the-Loop*, Renningen-Malmsheim : Expert-Verlag, Schriftenreihe des Instituts für Verbrennungsmotoren und Kraftfahrwesen der Universität Stuttgart, in German, 1997.
4. Rulka, W.: *Effiziente Simulation der Dynamik mechatronischer Systeme für industrielle Anwendungen*, Internal report IB 532–01–06, DLR Oberpfaffenhofen, Institut für Aeroelastik, in German, 2001.
5. García de Jalón, J. and Bayo, E.: *Kinematic and Dynamic Simulation of Multibody Systems: The Real-Time Challenge* Springer-Verlag New York, Inc., 1994.
6. Choi G.J. et al.: Vehicle Modeling Methods for Real-Time Dynamic Simulation Using Suspension Composite Joints, in *Mechanics of Structures and Machines*, Vol. 28, pp. 303–321, 2000.
7. Keßler, B.: *Bewegungsgleichungen für Echtzeitanwendungen in der Fahrzeugdynamik*, Institut B für Mechanik, Universität Stuttgart, in German, 1989.
8. Acevedo, M. and Celiqueta, J.T.: Real Time Dynamic Simulation of Passenger Cars, in *Mechatronics and Supercomputing Applications in the Transportation I*, pp. 559-566, 1994.

PARAMETRIC IDENTIFICATION OF THE ELASTIC POLE-VAULTING POLE

M.C. TOFAN, S.I. VLASE, I.I. MICU
University TRANSILVANIA
B-dul Eroilor, 29, Mecanica, 2200, Braşov, Romania

1. Introduction

Vaulting pole with elastic pole is a continuum communication between athletes and the pole during the vault. Kinetic energy introduced at the beginning of the vault can explain only 60-70 % of the total energy used to perform the vault. The rest, concerning 30-40 %, is introduced by the act of the athletes. In this study the authors wish to make an adequate mathematical model to explain the liaison and the power transfer between the vaulter and the elastic pole. The model contains a high elastic fiber-glass pole, free to pivot in the bottom end and charged with a variable momentum at the upper end. The inertia forces are also considered and all these forces subject the pole to a bending momentum and compression. The mathematical model is strong non-linear and for this reason involve some difficulties in solving the problem. This needs an adequate parameterization of deformed pole and the identification of the best modality to introduces power in the system.

In the paper is made an analysis of such complex motion and interaction between athletes and the pole in order to identify the parameters that describe the problem.

2. Mathematical Model

In the following we will present a proposed model in order to describe the motion of the vaulter. An analysis of the dynamic of the pole-vault event must include the effect of the highly elastic pole [1]. Hubbard [3],[4] proposed an iterative numerical solution, contending that an analytical solution was unknown. Griner [2] use the results proposed by Costello and Healey and offer a parametric solution to the pole-vault problem in terms of the tabulated elliptic integrals.

In this paper we use a vector representation [5] to describe the geometry of the non-linear pole in order to obtain, finally, the interaction between pole and pole-vaulter. We can write:

$$EI\frac{d\mathbf{r}}{ds} \times \frac{d^3\mathbf{r}}{ds^3} = \mathbf{F} \times \frac{d\mathbf{r}}{ds}$$

where:

$$\frac{d\mathbf{r}}{ds} \equiv \mathbf{t}$$

$$\frac{EI}{2}\left(\frac{d\theta}{ds}\right)^2 = F\,(q\cdot s(\theta)+c(\theta)+C) \xrightarrow{\theta=\alpha,\ \frac{d\theta}{ds}=\frac{1}{\rho}=0} C=-F(q\cdot s(\alpha)+c(\alpha))$$

The simpler model to describe the athlete is to consider that this is composed by two rigid body being in interaction with the pole. This description permit to obtain two kind of equations: one containing the elastic description and the other concerning the dynamic motion equations considering the two body that compose the athlete. For these two bodies we apply the well-known screw theorems. Between these two description exists a liaison made by some motion parameters. It is easy to describe the great deflection of the pole when is known the force and the torque at the end of the bar.

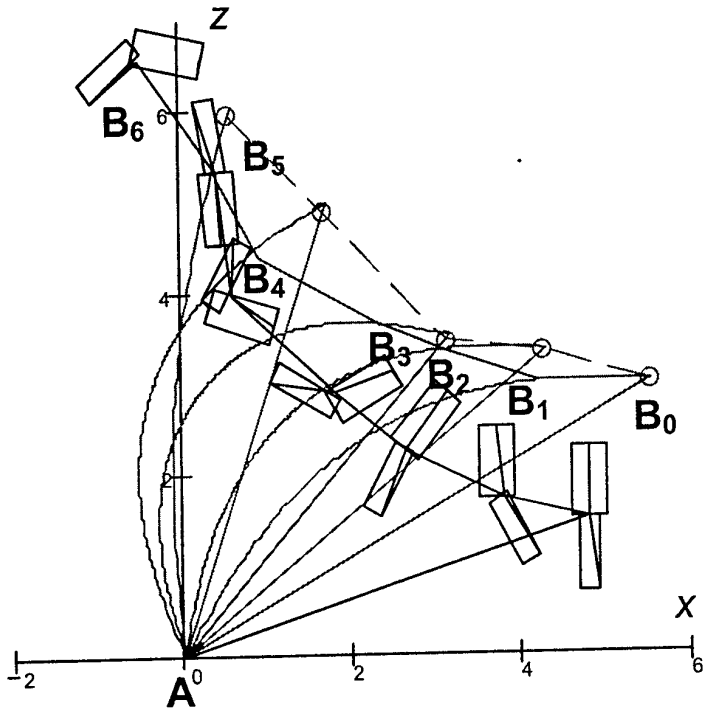


Figure 1. Seven successive positions of the vaulter

The most important things is to find the real values of the screw apply at the and of the pole at any moment of time. To determine this is necessary to consider for these two description two kind of differential equations: in one the variable is a coordinate that describe the pole and in the other set is the time. In every moment of time we must consider the solution of these two set of equations, the evolutions of the solutions being in a strong liaison.

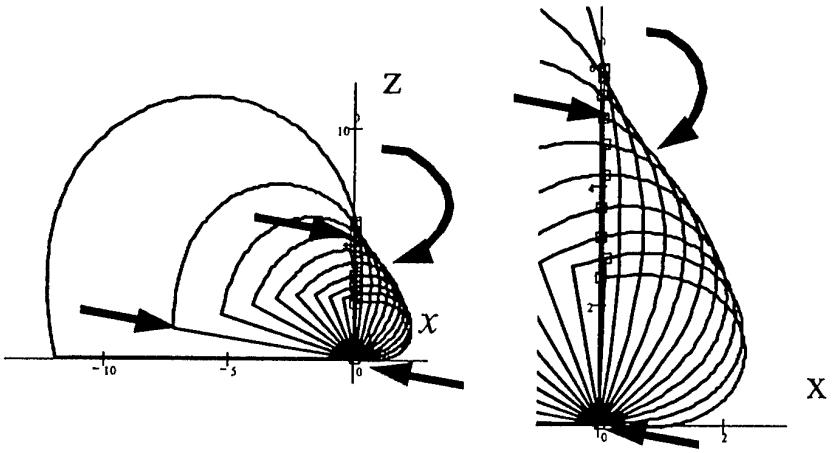


Figure 2. The large deflection of the pole in the Hubbard model

3. Experimental Results

Griner, in his paper, consider an experiment made by a vaulter. In this experiment he has recorded 65 of position. If we consider the athlete compose by two rigid bodies, seven position are presented in figure 1. He conclude that exists un vault which negative torque, at the end of the trajectory. The seven position are only a "frozen" sequence in a succession of all positions recorded.

Considering these results Griner perform a calculus of the large deflections considering the force and the torque adequate to obtain the experimental records. For some positions the screw considered to obtain the experimental results are not in a good accordance with the situation observed by the athletes. The discrepancy between the calculus and observations impose to consider the continuous liaison between athlete and pole.

4. Large Deflections of the Pole

In our researches we have considered an elastic pole being in a continuous "rigid" motion and have in the same time large deflections. The model has considered the interaction with the athlete acting at the end of the pole with a force screw (force and torque). For this model we have represented the elastic solutions for the succession of the seven positions considered.

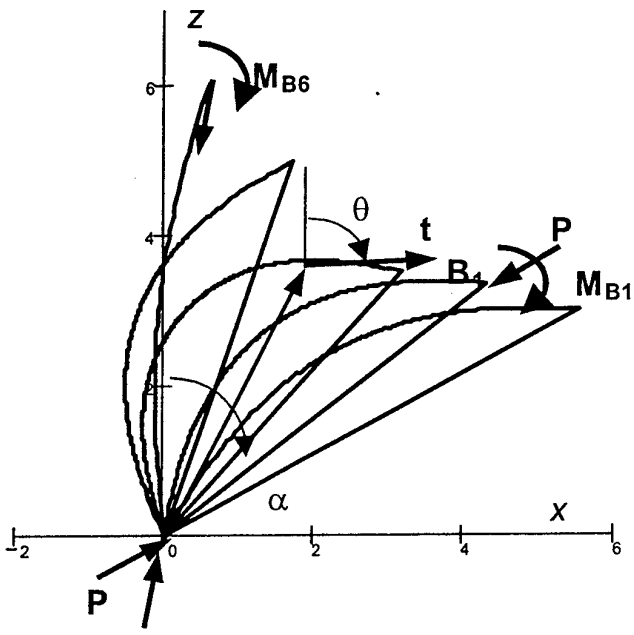


Figure 2. Large deflections of the pole

5. Conclusions

The results obtain by calculus and the experimental observations sustain the idea of a model of the elastic pole in an interaction with the vaulter. The motion of the vaulter is very complex and determine, in decisive manner, the deflection of the pole at any moment of time. The strategy of the vaulter is not only to transform the kinetic energy in a potential energy but too to use the arms to introduce a force screw in order to made higher the vault. To study the motion of the vaulter is necessary to consider the strong interaction that exists between the two parts of the system: the pole and the vaulter modeled by two rigid bodies. A good description of the motion is possible only in this case.

References

1. Costello, G.A. and Healey,J.J. (1964) Large Deflections of Elastic Columns with Various Supports. *Theoretical and Applied Mechanics*, Vol.2, Shaw, W.A.,ed., Pergamon Press, 527-543.
2. Griner, G.M. (1984) A parametric Solution to the Elastic Pole-Vaulting Pole Problem. ASME Journal of Applied Mechanics, Vol.51, 409-414.
3. Hubbard, M. (1980) Dynamics of the Pole Vault, *J. Biomechanics*, Vol.13, No.11, 965-976.
4. Hubbard, M. (1980) An Iterative Numerical Solution for the Elastica with Causally Mixed Inputs, ASME Journal of Applied Mechanics, Vol.47, 200-202.
5. Landau,L. and Lifchitz,E. (1967) Theorie de l'Elasticite, Editions MIR, Moscou, p.95-108.

DAViD - A MULTIBODY CODE TO SIMULATE A DYNAMIC VIRTUAL DUMMY FOR VIBRATIONAL COMFORT ANALYSIS OF CAR OCCUPANTS

P.P. VALENTINI
L. VITA
University of Rome "Tor Vergata" - Dept. of Mechanical Engineering
Via del Politecnico, 1 - 00133 - Rome - Italy
email: valentini@ing.uniroma2.it; vita@ing.uniroma2.it

1. Introduction

The development of virtual simulators can avoid to set-up expensive test rigs, time-consuming tests, and is a winning strategy to be more competitive in road-vehicles market. Moreover vibrational comfort analysis is an important topic in vehicle design and the possibility to perform virtual vibrational tests on the effects of changing some parameters is useful tool for the designer. A literature search reveals that most of the simulation models in the field are based on elementary linear models. In some cases, finite elements are used, but this approach involves a large amount of parameters to be defined and managed. Thus the authors of this paper developed a virtual dummy model by means of multibody techniques. The formulation is the one described in Haugh's text book [4]. The code, named DAViD (the acronym of Dynamic Automotive Virtual Dummy), can mimic the non linear behaviour of a 3D human body model and requires a very small set of body data. The model is completely parametric and can be automatically scaled to simulate a significant portion of population. The code can be also linked to experimental results of accelerometers time histories to perform multi-input analysis based on seat input (translational and rotational), steer wheel input and pedals input. Driver and occupants can be both simulated. It is possible to introduce non linear viscoelastic parameters to match the actual behaviour of cushion foams used in the manufacturing of seats. The model provides also an assessment of vibrational comfort computed in compliance with international standards. The results of the code DAViD have been compared with experimental ones acquired on a vibrational test rig.

2. Multibody Model

The developed model is based on a multibody dynamics approach [4]. In particular the whole model is made of 15 rigid elements, 12 of which define the dummy, and the remaining 3 describe the car environment. The dummy is composed of two feet, two legs, two thighs, the pelvis, two arms, two forearms an upper part that is formed by head, neck, shoulders and chest rigidly connected together. The other bodies included in the model are seat, pedals and steering wheel. In

order to represent the human body articulations, kinematics constraints and viscoelastic elements are used to connect each part of the dummy. There are two spherical joints between pelvis and thighs, two revolute joints with transverse axes between thighs and legs, two revolute joints with transverse axes between legs and feet, one prismatic joint with longitudinal axis between pelvis and upper part, two spherical joints between upper part and arms, two revolute joints with transverse axes between arms and forearms. The viscoelastic elements used in the dummy are one translational, between pelvis and the upper part to represent the stiffness of torso, and two rotational elements, between arm and forearm to reproduce the muscular elasticity of the elbow. The dummy interacts with the car environment by means of seat, pedal and steering wheel contact simulated by other viscoelastic elements. The contact between hands and steering wheel and feet and platform car is simulated with four very stiff springs. The model can automatically scale geometric, mass properties and spring locations by means of changing few parameters (such as percentile). In fact the code is interlaced with an anthropometrical database. It is also possible to modify the backrest inclination and the hip-heel vertical position in order to change the configuration of the seat. The code can also manage several inputs at the same time. It can get input acceleration time histories acquired by experimental tests, as well as time histories on velocities and positions, filtering the signals in order to suppress noise. If necessary, forces and torques could be introduced as well as driving constraints.

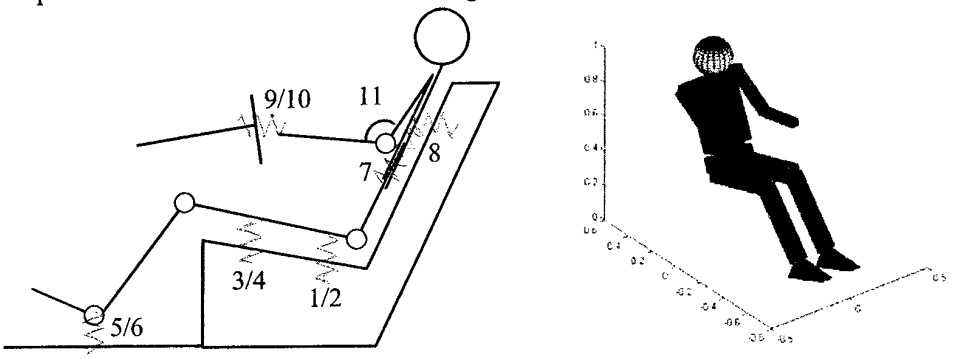


Figure 1. Viscoelastic elements (left) and complete 3-D dummy in Visualizer (right)

2.1 EQUATIONS OF THE MULTIBODY MODEL AND INTEGRATION

The equations of motion are deduced in the form of differential – algebraic system of index 3 [4] [5]:

$$\begin{cases} [M]\{\ddot{q}\} + [\Psi_q]^T \{\lambda\} = \{F_e\} \\ \{\Psi\} = \{0\} \end{cases} \tag{1}$$

where $[M]$ is the global mass matrix; $\{\Psi\}$ is the vector of constraint equation; $\{\lambda\}$ is the vector of Lagrangian multipliers; $\{F_e\}$ is the vector of external generalized forces; $\{q\}$ is the vector of generalized coordinates. In our model there are 15 bodies, and 105 generalized coordinates. The spatial location of the i-th body is described with seven parameters (i.e. three for the position of the

center of mass $q_{7i-6}, q_{7i-5}, q_{7i-4}$, and the four Euler's parameters q_{7i-3}, q_{7i-2}, q_{7i-1}, q_{7i}. The constraint equations used in the model can be divided into three groups:

- the first 15 equations (as many as the number of bodies in the model) are the normalization equations of the Euler's parameters (i.e. $q_{7i-3}{}^2 + q_{7i-2}{}^2 + q_{7i-1}{}^2 + q_{7i}{}^2 = 1$);
- the second group of equations is made up of the scleronomic constraints.
- the last group we include a driving constraint at inclination of pedals δ w.r.t. the horizontal plane (first two of (3)); regarding pelvis we impose no translation along z axis and rotation around the same axis (last three of (3)):

$$q_{39} = \cos\left(\frac{\delta}{2}\right);\ q_{46} = \cos\left(\frac{\delta}{2}\right);\ q_{17} = 0;\ q_{19} = 0;\ q_{20} = 0; \tag{3}$$

The complete model has 24 d.o.f. The integration of the DAE system, as it is shown in equations (1), has been performed rearranging the system as a first order one in the unique unknown y. Therefor the system to be integrated is in the following form:

$$[K]\{y'\} = \{\phi(y)\} \tag{4}$$

where:

$$[K] = \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};\ \{y\} = \begin{Bmatrix} q \\ \dot{q} \\ \ddot{q} \\ \lambda \end{Bmatrix} \text{ and } \{\phi(y)\} = \begin{Bmatrix} \{\dot{q}\} \\ \{\ddot{q}\} \\ \left\{[M]\{\ddot{q}\} + \left[\Psi_q\right]^T \{\lambda\} - \{F_e\}\right\} \\ \{\lambda\} \end{Bmatrix} \tag{5}$$

The system (5) is then solved by means of RADAU5.

3. Experimental set-up tests

A key point of simulation is the contact between seat and occupant that influences the vibrational response of the dummy. For this purpose an experimental procedure has been performed to find the seat force - deflection curve. Special mats, equipped with pressure transducers, are put on several seats and a jury made by people belonging to different physical groups has sat on the instrumented seat. Pressure maps have been acquired. Then spring elements have been introduced in the model and anchored to the points of high pressure concentration. For the computation of stiffness, appropriate tests have been performed on cushions using standard dynamometer. These have shown a non linear behaviour of polyurethane foams in their force/preload characteristic curves. A second kind of tests were performed to check the correct dynamic response of dummy. Some car have been tested on standard tracks and accelerometers signals have been acquired at measurement point (Figure 3). This signals have been replicated in a vibrational test rig, where the same seats have been mounted and the same driver has sat on. New signals have been acquired from SAE accelerometer pads placed on the cushion and on the backrest. The need for replicating these signals is due to obtain the repeatibility, and a standardization of the test procedures.

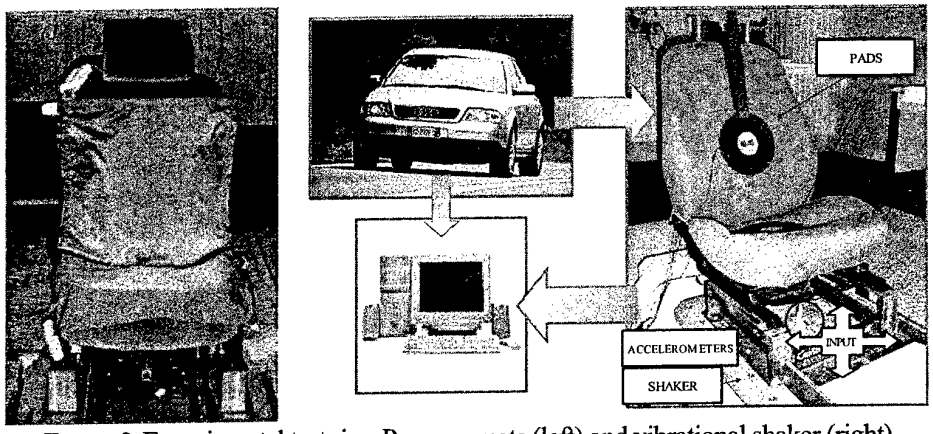


Figure 2. Experimental test rigs. Pressure mats (left) and vibrational shaker (right)

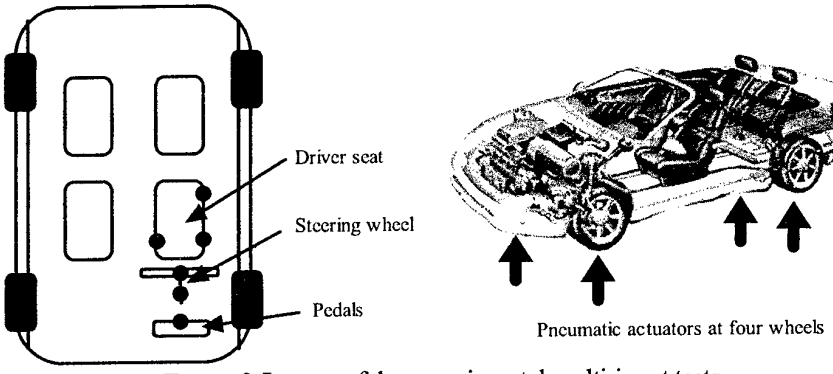


Figure 3. Layout of the experimental multi-input tests

4. Experimental multi-input tests

The DAViD code can take directly accelerometers data files to simulate a multi-input configurations The user can impose input at the seat (6 d.o.f.), at steer wheel (4 d.o.f.) and at pedals (3 d.o.f.). The signals can be pre-processed by filtering. It is possible to run analysis directly acquiring data from a four axis shaker experimental test rig on which a car has been placed. The pneumatic actuators reproduce the track profiles and give vibrational inputs to the tires, and the response signals at six accelerometers has been collected. The accelerometers, all with three axes of sensitivity have been placed as shown in Figure 3 three between the seat and the chassis, one on the steer column, one on the steer wheel, and one at the pedals.

5. Graphical User interface and 3-D Visualizer

The code DAViD is provided with a graphical user interface to simplify input phase. All the window interfaces have been developed using Visual Fortran. Many dialogs contain figures and

drawings that directly refer to model's parameters. The user can select percentile of occupant being simulated, parameters of posture, parameters for all spring – damper elements, kind of occupant (driver or passenger), input type, analysis parameters (format of files, simulation time, visualization steps) and external forces. Specific databases that contain information about anthropometrics of the subjects and elastic characterizations of the seats have been implemented in the model. The 3-D visualizer is an external code developed in Matlab language (Figure 1) that can be run after every analysis. The human body is represented using simple geometrical shapes. The visualizer is interlaced with the same percentile rescaling database of the DAViD code.

6. Comfort assessment and virtual perceiving

The comfort assessment is important to predict the effect provoked by vibrations on the human body. Many car accidents happens because of tiredness, or disturbs to perception, that can be avoided decreasing the level of transmitted vibration. Three aspects of vibration are fundamentals: the *exposure time*, the *amplitude* and the *frequency* [5]. The consequences of vibration exposure are not simple: the perception of motion, the sensations it produces and the interference with health and activities are all complex phenomena. Various standards for assessing whole-body vibration have been promulgated. These standards attempt to define easy methods of quantifying complex vibration conditions, nevertheless no simple standard can offer evaluation procedure which can accurately predict all known effects of vibration on the body. However, to estimate the comfort of car occupants, the authors have followed the method prompted by BS 6841 norm. According to such norm the Vibration Dose Value *VDV* is defined as follows:

$$VDV = \left[\int_{t=0}^{t=T} a_w{}^4(t)\,dt \right]^{1/4} \tag{6}$$

where $a_w(t)$ is the frequency-weighted acceleration time-history and T is the period of time over which vibration is measured. The evaluation of (6) requires the weighting of acceleration time history, that can be approximated, as stated in the norm, with piecewise functions. To compute the overall VDV the vibrational signals have to be measured at three points: seat cushion-body interface, seat backrest-body interface and ground-feet interface. These time histories are then frequency weighted and scaled with a factor variable from 0 to 1.0. For each weighted signals, VDV are computed and then an overall VDV is computed using:

$$VDV = \left(\sum_i VDV_i \right)^{1/4} \tag{7}$$

7. Results

In this section some results obtained running DAViD simulation code are presented and compared with those experimentally acquired. The simulated test is a multi-input one. The input data have been automatically filtered with a pass-band filter at the beginning of the run. The simulation time is 10 seconds. In Figure 4 are compared the FFT of the experimental and

computed time histories of pelvis (vertical acceleration) and upper part (horizontal acceleration). For the simulation we chose a driver belonging to 50 percentile, posed with the angle between legs and thighs of 105 deg, the head inclination of 18 deg, the neck inclination of 8 deg, the angle between the arms and the forearms of 42 deg, the pedals are at 20 deg.

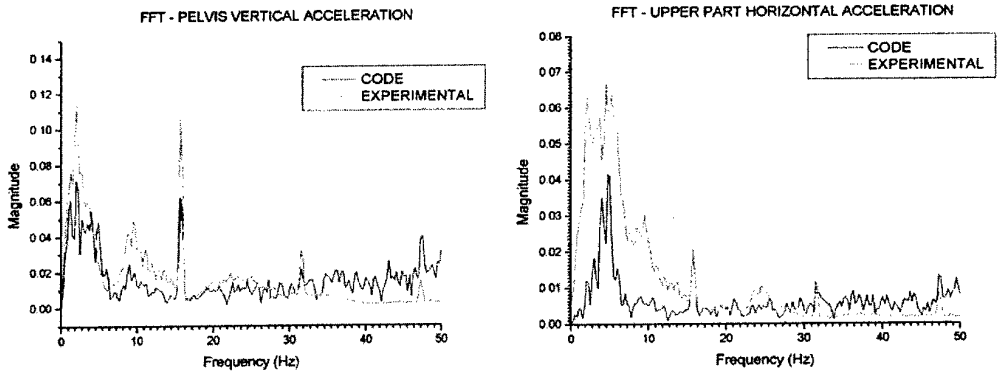


Figure 4. Comparison between computed and experimental FFTs

References

1.Griffin, M.J. (1990) *Handbook of Human Vibration*, Academic Press.

2. Silva, M.P.T., Ambrosio, J.A.C., Pereira, M.S. (1997) Biomechanical Model with Joint Resistance for Impact Simulation, *Multibody System Dynamics*, 65-84.

3. Kirchknopf, P. et al. Developement of a multibody calculation model for the passenger/seat system based on experimental results, *ATA Paper n.01A1085.*

4. Haugh, E. J.(1989) *Computer-Aided Kinematics and Dinamics of Mechanical Systems*, Allyn and Bacon, vol. I, pp.48-104.

5. Pennestrì, E. (2002) *Dinamica Tecnica e Computazionale*, Casa Editrice Ambrosiana, Vol. 2.

7. Lin Wei, Griffin, M. J. (9-2000) Effect of subject weight on predictions of seat cushion transmissibility, *35th United Kingdom Group Meeting on Human Responses to Vibration, ISRV*, University of Southampton, England.

9. Valentini, P.P. (A.Y. 1999-2000) Analisi delle modalità di implementazione di un manichino virtuale per la simulazione delle dinamiche vibrazionali del sistema uomo-sedile,Thesis, Univ. of Rome Tor Vergata.

10.Xuting Wu, Subhash Rakheia, Paul-Emile Boileau (1999) *Study of Human-Seat Interactions for Dynamic Seating Comfort Analysis*, Society of Automotive Engineers, Inc.

11.Yi Gu (1999) A New Dummy for Vibration Transmissibility Measurement in Improving Riding Comfort, Society of Automotive Engineers, Inc.

13. King,A.I.(5-1984)A Review of Biomechanical Models, *Journal of Biomechanical Engineering*, vol.106

14. Amirouche, F.M.L. Xie, M. Patwardhan, A. (11-1994)Optimization of the contact damping and stiffness coefficients to minimize human body vibration, *Journal of Biomechanical Engineering*, vol. 116.

15. Yi Gu (1988) *A Comparison Test of Trasmissibility Response from Human Occupant and Anthropodynamic Dummy*, Society of Automotive Engineers, Inc.

16. Vita L. (A.Y. 2000-2001) Sviluppo ed implementazione di un codice di calcolo 3D per lo studio della dinamica per applicazione all'interfaccia uomo/veicolo, Thesis, University of Rome Tor Vergata.

Acknowledgements The authors wish to thanks Prof. Ettore Pennestrì for his advice during this work and Ing. Federico Barizzone of Centro Ricerche Fiat for technical discussion.

DYNAMIC ANALISYS OF A LIGHT STRUCTURE IN SPACE: SHORT ELECTRODYNAMIC TETHER

J. VALVERDE, J.L. ESCALONA, J. MAYO, J. DOMÍNGUEZ
Department of Mechanical Engineering, University of Seville
Camino de los descubrimientos s/n 41092 Sevilla, Spain
e-mail: jvalverde@us.es

1. Introduction

A space tether is a long cable used to connect spacecrafts to one another or to other bodies (asteroids, space stations, boosters, etc.). One type of tether is the electrodynamic tether. This kind of tether interacts with the Earth's magnetic field producing a current in the tether itself by Faraday effect. According to the sense of the current, a Lorentz force appears on the tether, thrusting or dragging the motion of the system [1]. The stable position of an object orbiting in space is with its axis of smallest moment of inertia pointed towards the center of the Earth. This paper is a continuation of the work carried out by the authors [2] on a prototype of the European Space Agency (ESA) called SET (Short Electrodynamic Tether). As opposed to described earlier, the SET should orbit with its axis of smallest moment of inertia perpendicular to the plane of its orbit, see Figure 1.

The SET is composed of a central module from which two tethers extend, each about 100 meters in length. Before extension, the tethers are stored in drums. A plasma contactor can be found at the end of each tether. These contactors will be responsible for the emission and absorption of the electrons in the plasma for the production of the electric current mentioned earlier, [1]. Once extended, to maintain the operating position, an angular velocity ω_{spin} is applied to the SET around its axis of smallest moment of inertia. In this way, if the SET were rigid, the configuration would remain stable in the desired position by action of the gyroscopic pairs [3]. However, the SET is not rigid. On the contrary, it is a very flexible system that, as a result of the storage and extension of the tethers, will also not be perfectly rectilinear, see Figure 2. The system is therefore likened to a dynamics problem of unbalanced rotors. According to the literature on this subject [3], if the rotary system develops internal damping forces (hysteresis), there is a running speed called the "onset speed of instability" or critical velocity, above which the system becomes unstable. This onset speed of instability is practically equal to the first natural frequency of the rotor. In the case of the SET, its running speed ω_{spin} is much larger than its first natural frequency and there is internal damping. According to what has been stated earlier, it follows that transversal displacement of the contactor should grow without limits, making the SET unstable from a structural point of view. In the seventies, Genin & Maybee came to prove that if a non-linear model of internal damping and elastic forces is included in the equations of motion, the system is stable for any running speed ω [4]. In view of this, there is justification for a more detailed study of the dynamics of the SET. As opposed to the model carried out in [2], this study includes elastic and damping forces that retain second order and superior terms. As can be seen in the following section, the SET is modeled as a continuous system with a procedure belonging to multibody system dynamics. This is also something new in the field of tether dynamics. Figure 2 shows the model that has been solved, to which symmetry was applied with respect to the

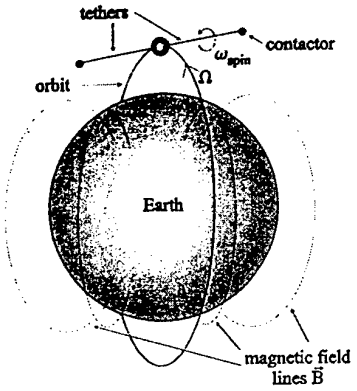


Figure 1. Orbit, orientation of the SET.

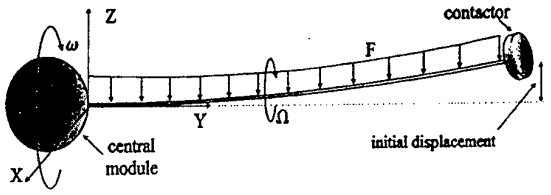


Figure 2. Model of the tether.

orbit plane. The angular velocity of the SET around the Earth Ω is much smaller than ω_{spin}, so that the axis fixed to the central module of the satellite can be considered inertial. According to [2], the electromagnetic forces (drag force) are modeled as a force distributed throughout the tether, which spins around it at velocity Ω. A study of this system may prove useful in future tether configurations that act in a similar way.

2. Modeling the problem; substructuring and natural coordinates

The problem to be solved will suffer large elastic displacements. Because of this, a substructuring method [5] was used, (Figure 3). The system will be modeled using the *floating reference frame approach* with natural coordinates as reference coordinates [6]. The use of this type of coordinates as coordinates of reference in a substructuring scheme produces a simple model of the system. This way, the constraints between the different substructures are automatically imposed and will not generate algebraic constraint equations associated to the rigid connections [6].

Consider the i-th substructure of the n in which the SET is divided. The set of coordinates $\{\mathbf{r}_0^i, \mathbf{u}_i, \mathbf{v}_i, \mathbf{w}_i\}$ defines the local axes that will be associated to the substructure and that are necessary for its kinematic description. The set of coordinates $\{\mathbf{r}_0^{i+1}, \mathbf{u}_{i+1}, \mathbf{v}_{i+1}, \mathbf{w}_{i+1}\}$ will therefore be in excess [6]. The position of any given point is

$$\mathbf{r}^i = \mathbf{r}_0^i + \mathbf{A}^i\left(\bar{\mathbf{r}}_u^i + \bar{\mathbf{u}}_f^i\right) \tag{1}$$

where vector $\bar{\mathbf{r}}_u^i$ represents the position of that point in local axes in the undeformed configuration and vector $\bar{\mathbf{u}}_f^i$ represents displacement due to deformation. The variables with a bar are expressed in local axes. The rotation matrix is $\mathbf{A}^i = [\mathbf{u}_i \quad \mathbf{v}_i \quad \mathbf{w}_i]$. The Rayleigh-Ritz method will be used in the discretization of the substructure. It should be noted that variations of the natural coordinates in excess $\{\mathbf{r}_0^{i+1}, \mathbf{u}_{i+1}, \mathbf{v}_{i+1}, \mathbf{w}_{i+1}\}$ will deform the substructure. Therefore, a Rayleigh-Ritz discretization with fixed boundaries will be carried out. This discretization will have a series of static modes, as well as dynamic modes with fixed boundaries (clamped-clamped beam) [6]. Thus,

$$\bar{\mathbf{u}}_f^i = \sum_{k=1}^{n_s} \bar{\Phi}_k \eta_k + \sum_{l=1}^{n_d} \bar{\Psi}_l \xi_l \tag{2}$$

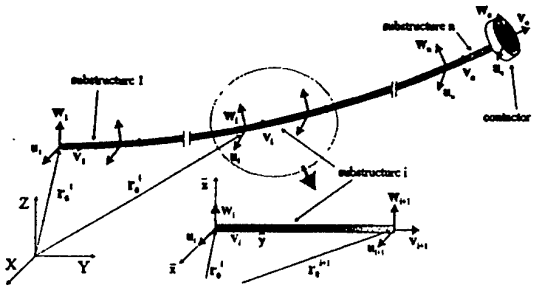


Figure 3. Substructuring and natural coordinates of the tether

where n_s and n_d are the number of static and dynamic modes respectively; $\overline{\Phi}_k$ is a 3x1 vector that contains the static mode in the corresponding row, according to whether displacement is produced in $\bar{x}$, $\bar{y}$ o $\bar{z}$, $\overline{\Psi}_l$ is a vector containing dynamic modes in the same way, and η_k and ξ_l are the amplitudes of the static and dynamic modes respectively. The vector of coordinates for the i-th substructure will therefore be

$$\mathbf{q}^i = \left[\mathbf{r}_0^{iT} \quad \mathbf{u}_i^T \quad \mathbf{v}_i^T \quad \mathbf{w}_i^T \quad \eta_1 \quad \cdots \quad \eta_{n_s} \quad \xi_1 \quad \cdots \quad \xi_{n_d}\right]^T \tag{3}$$

The expression of the elastic forces [6] of a substructure is obtained from the vector

$$\overline{\mathbf{u}}^i = \overline{\mathbf{r}}_u^i + \overline{\mathbf{u}}_f^i = \left[\overline{u}_x \quad \overline{u}_y \quad \overline{u}_z\right]^T \tag{4}$$

supposing that it behaves as an Euler-Bernoulli beam.

The inertia forces will be obtained using the *co-rotational method* proposed by Géradin and Cardona, adapted to the use of natural coordinates in [7]. This is way to obtain a much simpler expression of the inertia forces than by deriving expression (1). Consider the i-th substructure of the SET, see Figure 3. If the substructure is divided in $p-1$ finite elements, the following interpolation in velocities can be made

$$\dot{\mathbf{r}} = \mathbf{N}\mathbf{v}^* \tag{5}$$

where $\mathbf{v}^* = \left[\mathbf{v}^{1T} \quad \cdots \quad \mathbf{v}^{jT} \quad \cdots \quad \mathbf{v}^{pT}\right]^T$ is the derivative with respect to the time of the nodal displacements in global coordinates (i superscript removed for simplicity).

If a linear interpolation is considered for expression (5), there will be a good approximation of the velocities of the body in relation to the velocity of the nodes. Then, linear finite elements will be used. The following expression can be arrived at by using expression (5) to obtain the kinetic energy of a substructure

$$T = \tfrac{1}{2}\int_V \dot{\mathbf{r}}^T\dot{\mathbf{r}}\,dm = \tfrac{1}{2}\mathbf{v}^{*T}\mathbf{M}_{MEF}\mathbf{v}^* \tag{6}$$

where $\mathbf{M}_{MEF}$ is the mass matrix that appears in the *finite element method*. Expression of kinetic energy is very simple, but it is not expressed in relation to the coordinates of the substructure $\mathbf{q}$, see (3). It is necessary to find a relation between $\mathbf{v}^*$ and $\dot{\mathbf{q}}$. The following is reached by deriving (1) and equaling (5), particularizing at the nodes

$$\mathbf{v}^* = \dot{\mathbf{r}}_0 + \dot{\mathbf{A}}\left(\overline{\mathbf{r}}_u + \overline{\mathbf{u}}_f^*\right) + \mathbf{A}\dot{\overline{\mathbf{u}}}_f^* = \mathbf{B}(\mathbf{q})\dot{\mathbf{q}} \tag{7}$$

where $\overline{\mathbf{u}}_f^*$ is obtained particularizing expression (2) at the nodes. $\mathbf{B}(\mathbf{q})$ [7] is the simple matrix that relates $\mathbf{v}^*$ and $\dot{\mathbf{q}}$. Thus, the kinetic energy is

$$T = \tfrac{1}{2}\dot{\mathbf{q}}^T\mathbf{B}^T\mathbf{M}_{MEF}\mathbf{B}\dot{\mathbf{q}} = \tfrac{1}{2}\dot{\mathbf{q}}^T\mathbf{M}\dot{\mathbf{q}} \tag{8}$$

In (8), the kinetic energy is expressed in relation to $\dot{\mathbf{q}}$. There is an adequate approximation to the inertia forces and a simple expression of the mass matrix has been achieved. Furthermore, the expression of the quadratic velocity vector $\mathbf{Q}_v$ will be

$$\mathbf{Q}_v = -\mathbf{B}^T \mathbf{M}_{MEF} \dot{\mathbf{B}} \dot{\mathbf{q}} \tag{9}$$

where the calculation of $\dot{\mathbf{B}}$ is quite simple [7].

The hysteretic damping of the material plays an essential role in the stability of the system. It has been modeled as viscous damping which introduces the same destabilizing effect as hysteretic damping [3]. Internal viscous damping was supposed as proportional to modal mass and stiffness.

The vector of external forces associated to electromagnetic force (distributed force), see Figure 2, will be obtained from the following expression

$$\mathbf{Q}_{ext} = \int_0^l \mathbf{F}^T \frac{\partial \mathbf{r}}{\partial \mathbf{q}} dy \tag{10}$$

where $\mathbf{r}$ is given by (1) and $\mathbf{q}$ by (3). $\mathbf{F}$ will be expressed in global axes as

$$\mathbf{F}^T = [F\cos(\Omega t) \quad 0 \quad F\sin(\Omega t)] \tag{11}$$

Once the equations of the different substructures have been put together and the pertinent constraints have been imposed, the resulting system of equations will be

$$\mathbf{M}\ddot{\mathbf{q}} + \Phi_{\mathbf{q}}^T \boldsymbol{\lambda} = \mathbf{Q}\,;\, \Phi(\mathbf{q}, t) = \mathbf{0} \tag{12}$$

where $\mathbf{M}$ is the mass matrix of the complete system, $\mathbf{q}$ is the vector of coordinates in the problem, Φ are the constraints, and $\mathbf{Q}$ is a vector containing all the forces acting on the system [6].

In order to solve the equations of motion, an index-3 Lagrangian method with mass-orthogonal projections of the velocities and accelerations to their constraint manifolds was used. This formulation is proposed by Bayo et al and revised in [7].

3. Results

Due to the difference between the frequencies associated to longitudinal and transversal motions, the system of differential-algebraic equations is stiff. Without lost of generality, a longitude of $L = 30m$ is used to integrate the system of equations (12). This reduces the size of the problem and the time of computation. The section of the tether, hollow and thin-walled, has the following properties: $EI_x = 38\,Nm^2$, $EI_z = 21\,Nm^2$, area $A = 4.16\times10^{-6}\,m^2$. The tether is made of a copper-beryllium alloy with the following properties [2]: Young modulus $E = 132\,GPa$, Poisson ratio $\nu = 0.3$, and internal damping constant $\xi = 0.05$. The plasma contactor, with a mass of 3 Kg., was modeled as a disk with a radius of 0.2 m and thickness of 0.01 m. The electromagnetic force will be a distributed force with value $F = 40\times10^{-6}\,N/m$. The tether is initially deformed with a displacement of the contactor about $0.1\,L$. The running speed will be applied in the central module.

Following are the results obtained for in three cases: linear model $n = 1$ (n = number of substructures) and non-linear models $n = 3$ and $n = 6$. The critical velocity of the system will be close to its first natural bending frequency $\omega_1 \approx 0.03 rad/s$. The problem has been solved for subcritical $\omega = 0.02\,rad/s$ and supercritical

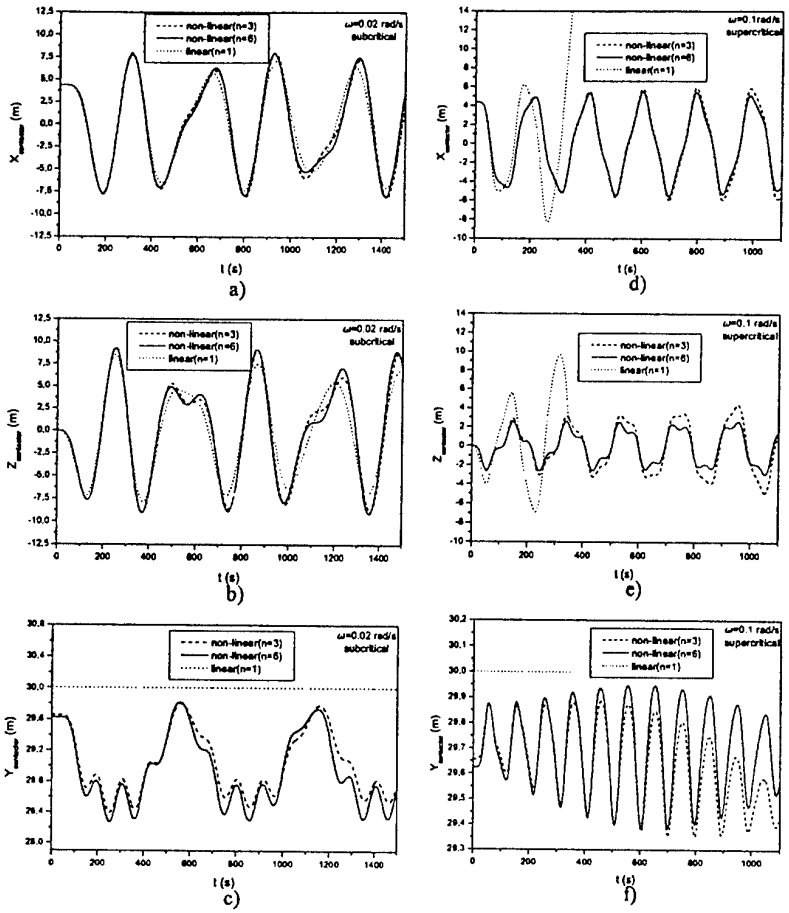


Figure 4. Evolution of the contactor for sub and supercritical velocities

$\omega = 0.1\,rad/s$ velocities, starting from zero, gradually reaching the desired velocity in 10 seconds. Each substructure was divided into 10 finite elements to carry out the approximation of the inertia forces (see (5)), and 6 dynamic modes were taken. This case requires 36 ($n=1$), 108 ($n=3$) and 216 ($n=6$) coordinates for the description of the problem. This gives an idea of the size of the problem and how it grows with an increase in the number of substructures.

Figure 4 shows the evolution of the contactor projected on global axes (see Figure 2) for each of the cases analyzed. Figures 4a) and 4b), show how, for subcritical velocity, the displacement developed by the contactor remains stable under the value of 8 m in direction X ($0.26L$) and 10 m in direction Z ($0.33L$). This is the case for both the linear model and the non-linear models. Figures 4c) and 4f) show how for the linear model there is no shortening of the tether in direction Y, in spite of the fact that displacement of the contactor is different from zero at all moments. This is because in the linear model, the bending and tensile-compressive forces are uncoupled and there are no forces to excite the axial modes. However, in the non-linear model, an increase in the X-Z displacement of the contactor should translate into a decrease in the Y displacement of the contactor. The effect produced by this coupling can be observed in

Figures 4a) and 4b) in $t = 550s$ and $t = 1150s$, where linear and non-linear solutions are separated. Figures 4d), 4e), and 4f) represent the evolution of the contactor when the SET is submitted to supercritical velocity. In the linear model, displacement grows rapidly. When displacements are so large (14 m, large elastic displacements) that the linear model is no longer valid, the integrator does not converge and it comes to a stop. In any case, it is understood that the motion is not stable. On the other hand, motion does remain stable in the non-linear model under the value of 5 m ($0.16L$) in direction X and 3 m ($0.1L$) in direction Z. The non-linearities in elastic and internal damping forces seem to stabilize the motion of the SET as predicted in [4].Figure 4 reveals the influence of the number of substructures into which the tether is divided. Figures 4d) and 4e) indicate how displacement for $n = 6$ is slightly smaller than for $n = 3$. The larger the number of substructures, the larger the coupling between transversal and longitudinal displacements, so that non-linear behavior of the system is simulated with greater precision. This explains why displacement for $n = 6$ is smaller than for $n = 3$. Moreover, for $n = 3$ displacement is slowly destabilized; this does not occur for $n = 6$. The solution for $n = 6$ was considered definite, as it practically coincides with solutions found for $n > 6$.

4. Conclusions and future projects

This paper carried out a non-linear model of the SET in elastic and internal damping forces. The solutions obtained suggest a stable behavior of the system, both at supercritical and subcritical velocities. In any case, an in-depth study of the stability of the system in order to determine the values of the parameters that characterize the movements of the SET for which the system behaves in a stable manner is justified. Once these areas of stability are known it will be much easier to work with the SET, since prior knowledge of the behavior of the system will be helpful for the analyst.

References

[1] Cosmo, M. and Lorenzini, C. (1997) *Tethers in space handbook*, Smithsonian Astrophysical Observatory (NASA Marshall Space Flight Center), Third Edition.

[2] Mayo, J.,Martínez, J., Escalona, J.L. and Domínguez, J. (1999) *Short electrodynamic tether WP-200 Design and Mechanism Final report*, Dep. de Ingeniería Mecánica y de los Materiales, Universidad de Sevilla.

[3] Childs, D. W. (1993) *Turbomachinery Rotordynamics*, Wiley Interscience.

[4] Genin, J. and Maybee, J. S. (1969) Stability in the three dimensional whirling problem, *International Journal of Non-linear Mechanics*, **4**, 205-215.

[5] Wu, S. and Haug, E. (1988) Geometric non-linear substructuring for dynamics of flexible mechanical systems, *International Journal for Numerical Methods in Engineering*, **26**, 2211-2226.

[6] Cuadrado, J., Cardenal, J. and García de Jalón, J. (1996) Flexible mechanisms through natural coordinates and component synthesis: an approach fully compatible, *International Journal for Numerical Methods in Engineering*, **39**, 3535-3551.

[7] Cuadrado, J., Gutiérrez, R., Naya, M.A. and Morer, P. (2001) A comparison in terms of accuracy and efficiency between a MBS dynamic formulation with stress analysis and a non-linear FEA code, *International Journal for Numerical Methods in Engineering*, **51** (9), 1033-1052.

SOME ASPECTS OF FINITE ELEMENT ANALYSIS OF FLEXIBLE MULTI-BODIES SYSTEMS

S.I. VLASE, M.C. TOFAN, I.A. GOIA
University TRANSILVANIA
B-dul Eroilor, 29, Mecanica, 2200, Braşov, Romania

1. Introduction

In many cases when a study of a multi-bodies system is perform, the basic hypothesis used is that all elements are rigid. In reality the elasticity of the components can be large enough so that the dynamic response can be not only quantitative but also qualitative different. For this reason, in some applications, particularly in the field of robotics and high-speed vehicles, is necessary to consider the elasticity of elements and to use correspondent models. Generally, the multi-bodies systems have a great complexity and a strong non-linearity. To study such systems with the classic mechanics theorems is not a practical task because the motion equations have, generally, no analytical solutions. For this reason is necessary to use numerical methods and the finite element methods (FEM) remains one of the most important tools [1]-[4],[6]-[8],[9],[12].

The major difficulty using FEM is the non-linearity of the motion equations. The coefficients that appears in equations are position (time) dependent and, in some practical application (mechanisms with a periodical motion) they can be period. To solve this problem the motion must be considered "frozen" for a very short interval of time. In this case the obtained equations can be considered linear.

It exists two difficult and major problems when is used finite element method: one consist in the fact that the equations contain more terms as in the classical procedures and the second is that the equations are only incremental valid, for a very short time interval; after this interval must generate new coefficient for the motions equations and the solutions previously obtained are the initial conditions for the new equations.

In the paper are established the incremental motion equations for a general multi-bodies system with elastic elements being in a three-dimensional motion and are analyzed the problems involved using FEM procedures.

2. Motion Equations

In the following we will establish the motion equations for an elastic finite element with a general motion together with an element of the system. The type of the shape function is determined by the type of the finite element. For this reason we will present the motion equations in three different situations: for a three-dimensional finite element with

a general three-dimensional motion, for a two-dimensional finite element with a plane motion and for an one-dimensional element with a general three-dimensional motion. We will consider that the small deformations will not affect the general, rigid motion of the system.

We consider that, for the all elements of the system, we know the field of the velocities and of the accelerations. We refer the finite element to the local coordinate system Oxyz, mobile, and having a general motion with the part of system considered (fig.1). We note with $\vec{v}_o(\dot{X}_o,\dot{Y}_o,\dot{Z}_o)$ the velocity and with $\vec{a}_o(\ddot{X}_o,\ddot{Y}_o,\ddot{Z}_o)$ the acceleration of the origin of the local coordinate system. The motion of the whole system is refer to the general coordinate system O'XYZ. By [R] is denoted the rotation matrix.

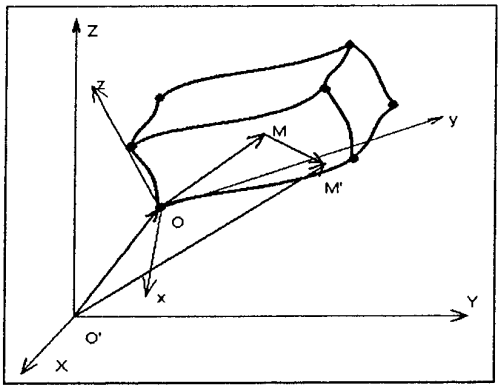


Figure 1. Finite element in a three-dimensional motion

We note by $\{r'\}$ the position vector MM' with the components in the general coordinate system O'XYZ. The point M has a displacement $\{f\}$ and become M':

$$\{r_{M'}\}=\{r_o\}+[R](\{r'\}+\{f\}) \tag{1}$$

where $\{r_{M'}\}$ is the position vector of point M' with the components express in the global reference system. The continuous displacement field $\{f(x,y,z,t)\}$ is approximated, in FEM, by:

$$\{f\}=[N(x,y,z)]\{\delta_e(t)\} \tag{2}$$

where the elements of matrix [N] (the shape functions), are determined by the type of the finite element choose.

The velocity of point M' will be:

$$\{v_{M'}\}=\{\dot{r}_o\}+[\dot{R}]\{r'\}+[\dot{R}]\{f\}+[R]\{\dot{f}\}=\{\dot{r}_o\}+[\dot{R}]\{r'\}+[\dot{R}][N]\{\delta_e\}+[R][N]\{\dot{\delta}_e\} \tag{3}$$

The kinetic energy of the finite element considered is:

$$E_c=\frac{1}{2}\int_V \rho v^2 dV=\frac{1}{2}\int_V \rho\{v_{M'}\}^T\{v_{M'}\}dV \tag{4}$$

The relations between strains and finite deformations are $\{\varepsilon\}=[a]\{f\}$ where [a] is a differentiation operator and the deformation energy is:

$$E_p = \frac{1}{2}\int_V \{\delta_e\}^T [k_e]\{\delta_e\} dV \qquad (56)$$

where $[k_e]$ is the rigidity matrix for the e element:

$$[k_e] = \int_V [N]^T [a]^T [D]^T [a][N] dV \quad . \qquad (6)$$

If we not with $\{p\} = \{p(x,y,z)\}$ the distributed forces vector, the external work of these is:

$$W = \int_V \{p\}^T \{f\} dV = \left(\int_V \{p\}^T [N] dV \right) \{\delta_e\}. \qquad (7)$$

and the nodal forces $\{q_e\}$ produce an external work:

$$W^c = \{q_e\}^T \{\delta_e\} \qquad (8)$$

The Lagrangean for the considered element is obtain with the relation:

$$L = E_c - E_p + W + W^c \qquad (9)$$

If we apply the Lagrange's equations after some algebraic operations we obtain the motion equations for a single finite element under the form:

$$\left(\int_V [N]^T [N] \rho dV \right) \{\ddot{\delta}_e\} + 2\left(\int_V [N]^T [R]^T [\dot{R}][N] \rho dV \right) \{\dot{\delta}_e\} +$$

$$+ \left([k_e] + \int_V [N]^T [R]^T [\ddot{R}][N] \rho dV \right) \{\delta_e\} =$$

$$= \{q_e\} + \int_V [N]\{p\} dV - \left(\int_V [N]^T \rho dV \right) [R]\{\ddot{r}_o\} - \int_V [N]^T [R]^T [\ddot{R}]\{r'\} \rho dV \qquad (10)$$

With the notations:

$$\{m_{ix}\} = \int_V [N_{(i)}]^T x \rho dV \quad ; \quad \{m_{iy}\} = \int_V [N_{(i)}]^T y \rho dV ; \quad \{m_{iz}\} = \int_V [N_{(i)}]^T z \rho dV \quad ;$$

$$[m_{ij}] = \int_V N_{(i)} N_{(j)}^T \rho dV \quad ; \quad \{q^*\} = \int_V [N]^T \{p\} dV \quad ; \quad [m_{oe}^i] = \int_V [N]^T \rho dV \quad ;$$

$$[m_e] = [m_{11}] + [m_{22}] + [m_{33}] \quad ; \quad \{q_e^i(\Omega)\} = \int_V [N]^T [\Omega][\Omega]\{r'\} \rho dV$$

$$[c_e(\Omega)] = \int_V [N]^T [\Omega][N] \rho dV \quad ; \quad [k_e(E)] = \int_V [N]^T [E][N] \rho dV$$

$$[k_e(\Omega^2)] = \int_V [N]^T [\Omega][\Omega][N] \rho dV \quad ; \quad \{q_e^i(E)\} = \int_V [N]^T [E]\{r'\} \rho dV$$

it result the motion equations for the finite element analyzed, where $\vec{\Omega}$ represent the angular velocity and $\vec{E}$ the angular acceleration with the components in the local coordinate system:

$$[m_e]\{\ddot{\delta}_e\} + 2[c_e]\{\dot{\delta}_e\} + ([k_e] + [k_e(E)] + [k_e(\Omega^2)])\{\delta_e\} =$$

$$= \{q_e\} + \{q_e^*\} - \{q_e^i(E)\} - \{q_e^i(\Omega^2)\} - [m_{oe}^i][R]^T \{\ddot{r}_o\} \qquad (11)$$

These motion equations are referred to the local coordinate system and the nodal displacement vector $\{\delta_e\}$ and the nodal force vector $\{q_e\}+\{q_e^*\}$ are express in the same coordinate system. The motion equations are true for the instantaneous position of the system. We consider that the system is „frozen" for the moment considered. The expression (12) contain some remarkable terms:

- $2[c_e]\{\dot{\delta}_e\}$ - represent the accelerations of Coriolis type and the cause of these is the relative velocity $\{\dot{\delta}_e\}$ of the nodal coordinates;
- $[k_e(E)]+[k_e(\Omega^2)]$ - modify the stiffness matrix and the cause are the relative motion express by the angular velocity and angular acceleration;
- $\{q_e^i(E)\}+\{q_e^i(\Omega^2)\}$ - represent the inertia effects due to the rotation of the local coordinate system;
- $[m_{oe}^i][R]^T\{\ddot{r}_o\}$ - represent the inertia effects due to the translation of the finite element.

When the two-dimensional finite element is in a plane motion, if we use the same procedures, we obtain the motion equations for this case:

$$[m_e]\{\ddot{\delta}_e\}+2[c_e]\{\dot{\delta}_e\}+([k_e]+\varepsilon[k_e]-\omega^2[m_e])\{\delta_e\}=$$
$$=\{q_e\}+\{q_e^*\}-\{q_e^i(E)\}-\{q_e^i(\Omega^2)\}-[m_{oe}^i][R]^T\{\ddot{r}_o\} \qquad (12)$$

In the case of a one-dimensional finite element, there exists some special forms for the deformation energy. We must take into account that the second order effects make more stiff the element, when this perform a motion with a high speed. Finally we can obtain for this situation the motion equations:

$$[m_e]\{\ddot{\delta}_e\}+2[c_e]\{\dot{\delta}_e\}+([k_e]+[k_e(E)]+[k_e(\Omega^2)]+[k_e^G])\{\delta_e\}=$$
$$=\{q_e\}+\{q_e^*\}-\{q_e^i(E)\}-\{q_e^i(\Omega^2)\}-[m_{Ee}^i][I]\{E\}-[m_{oe}^i][R]^T\{\ddot{r}_o\} \qquad (13)$$

where matrix $[k_e^G]$ take into account the second order effects and the term $[m_{Ee}^i][I]\{E\}$ describe the influence of the rotation inertia. The shape functions will determine the final form of the matrix considered in these equations.

3. Assembling Procedures and Liaison Forces Eliminating

3.1. KINEMATICS

In the following the authors present an analytic method to justify the assembling methods used for this type of systems. The unknowns in the elasto-dynamic analysis of a mechanical system with liaisons are the nodal displacements and the liaison forces. By assembling the motion equations written for each finite element we try to eliminate the liaisons forces and the motion equations will contain only nodal displacements as unknowns. The liaison between finite elements are realized by the nodes where the displacements can be equal or can exists other type of functional relations between these. When two finite elements belong to two different elements (bodies) the liaison realized by node can imply relations more complicated between nodal displacement and their

derivatives. Generally, the relations between the first order derivative of the nodal displacements can be expressed by the linear formulas:

$$\{\dot{\Delta}\} = [A]\{\dot{q}\} \tag{14}$$

where by $\{\Delta\}$ we have noted the nodal displacement vector and by $\{q\}$ the nodal independent displacements. By differentiation (14) we obtain:

$$\{\ddot{\Delta}\} = [A]\{\ddot{q}\} + [\dot{A}]\{\dot{q}\} \tag{15}$$

The transformation relations between the displacements expressed in the global fix coordinate system $\{\Delta_e\}$ and the displacements expressed in the local mobile coordinate system $\{\delta_e\}$ are:

$$\{\Delta_e\} = [R_e]\{\delta_e\} \tag{16}$$

where index e denote the e-th element.

3.2. DYNAMIC SYSTEM DESCRIPTION

For a single finite element that belong to an elastic component of the system that has a general three-dimensional rigid motion with the angular velocity $\vec{\omega}$ and the angular acceleration $\vec{\varepsilon}$ (or $\vec{\Omega}$ and $\vec{E}$ in the mobile coordinate system) we consider the motion equations obtained by the relation (11). For the other cases the procedures are the same.

The equations are expressed in the local mobile reference system. If we write these equations in the global fix coordinate system, they keep there form:

$$[M_e]\{\ddot{\Delta}_e\} + 2[C_e]\{\dot{\Delta}_e\} + \left([K_e] + [K_e(\varepsilon)] + [K_e(\omega^2)]\right)\{\Delta_e\} =$$
$$= \{Q_e\} + \{Q^*_e\} - \{Q^i_e(E)\} - \{Q^i_e(\Omega^2)\} - [M^i_{oe}][R]^T\{\ddot{r}_o\} \tag{17}$$

We will note in the following:

$$\{Q_e\}^{inertia} = -\{Q^i_e(E)\} - \{Q^i_e(\Omega^2)\} - [M^i_{oe}][R]^T\{\ddot{r}_o\}$$

and we can obtain finally the motion equations for the whole structure, referred to the global coordinate system, under the form:

$$[M]\{\ddot{\Delta}\} + 2[C]\{\dot{\Delta}\} + \left([K] + [K(\varepsilon)] + [K(\omega^2)]\right)\{\Delta\} = \{Q\}^{ext} + \{Q^*\}^{ext} + \{Q\}^{leg} + \{Q\}^{inertie} \tag{18}$$

If we take into account the relations (18) and (20) we can write:

$$[M]\ \left([A]\{\ddot{q}\} + [\dot{A}]\{\dot{q}\}\right) + 2[C][A]\{\dot{q}\} + \left([K] + [K(\varepsilon)] + [K(\omega^2)]\right)[A]\{q\} =$$
$$= \{Q\}^{ext} + \{Q^*\}^{ext} + \{Q\}^{leg} + \{Q\}^{inertie} \tag{19}$$

3.3. WORK OF LIAISON FORCES

It can be shown [10], [11] that the work of the liaison forces for system can be written:

$$dL = \{\dot{\Delta}\}^T\{Q\}^{leg}\,dt = \{\dot{q}\}^T[A]^T\{Q\}^{leg}\,dt \tag{21}$$

But the work due to the liaison forces is null for an ideal system [5], [14] and the independence of the nodal coordinates q offer the relation:

$$[A]^T\{Q\}^{leg} = 0 \tag{22}$$

that is the basic relation in the following.

3.4. MOTION EQUATIONS ASSEMBLING

We consider relation (19) and we pre-multiply this with $[A]^T$. It result:

$$[A]^T[M][A]\{\ddot{q}\}+([A]^T[M][\dot{A}]+2[C][A])\{\dot{q}\}+[A]^T([K]+[K(\varepsilon)]+[K(\omega^2)])[A]\{q\}=$$
$$=[A]^T\{Q\}^{ext}+[A]^T\{Q^*\}^{ext}+[A]^T\{Q\}^{leg}+[A]^T(\{Q^*\}^{leg}+\{Q\}^{inertie}) \quad (23)$$

If we take into account the relation (22) the liaison forces (the nodal forces) vanish and it result a system of equations without liaison forces and the unknown are only the nodal displacements. This result justify the assembling methods used in the case of the mechanical systems with liaisons analyzed via finite element method.

$$[A]^T[M][A]\{\ddot{q}\}+([A]^T[M][\dot{A}]+2[C][A])\{\dot{q}\}+[A]^T([K]+[K(\varepsilon)]+[K(\omega^2)])[A]\{q\}=$$
$$=[A]^T\{Q\}^{ext}+[A]^T\{Q^*\}^{ext}+[A]^T\{Q\}^{inertia} \quad (24)$$

The system of differential equations obtained is nonlinear, the matrix of the left term depending on the configuration of the multi-body system. These equations contain the "rigid motion" of the system and for these they have one or more singularities. To solve the equations the rigid motion must be eliminated.

References

1. Bahgat, B.M. and Willmert, K.D. (1976) Finite Element Vibrational Analysis of Planar Mechanisms. *Mechanis and Machine Theory*, vol.11, 47-57.

2. Blejwas, E.T. (1981), The Simulation of Elastic Mechanisms Using Kinematics Constraints and Lagrange Multipliers. *Mechanism and Machine Theory*, vol.16, Nr.4, 425-431.

3. Cleghorn, W.L., Fenton, E.G. and Tabarrok, K.B. (1981) Finite Element Analysis of High-Speed Flexible *Mechanisms. Mechanism and Machine Theory*, vol.16, Nr.4, 407-413.

4. Erdman, A.G., Sandor, G.N., Oakberg, A.(1972) A General Method for Kineto- elasto-dynamic Analysis and Synthesis of Mechanisms. *Journal of Engineering for Industry*, ASME Trans., November, 1193-1199.

5. Iacob,C. (1980) Mecanica teoretica, Ed. did. si ped., Bucuresti,.

6. Mifha, A.(1978) Finite Element Approach to Mathematical Modeling of High Speed Elastic Linkage. Mechanisms and Machine Theory, vol.13, 603-609.

7. Mocanu, D., Goia, I., Vlase, S. and Vasu, O. (1990) Experimental Checkings in the Elasto-dynamic analysis of mechanism, by using finite elements. *International Congress On Experimental Mechanics*, Lyngby, Denmark, 1053-1058.

8. Nath, P.K. and Ghosh, A.(1980) Kineto- Elastodynamic Analysis of Mechanism by Finite Element Method. *Mechanism and Machine Theory*, vol. 15, 179-189.

9. Vlase, S.(1985) Elastodynamische Analyse der Mechanischen Systeme durch die Methode der Finiten Elemente. *Bul. Univ. Brasov*, p.1-6.

10. Vlase, S.(1987) A Method of Eliminating Lagrangean Multipliers from the Equations of Motion of Interconnected Mechanical Systems. *Journal of Applied Mechanics*, ASME trans., vol.54, nr.1,

11. Vlase, S. (1987) Elimination of Lagrangian Multipliers. Mechanics Research Communications,vol. 14, 17-22.

12. Vlase, S. (1994) Modeling of Multibody Systems with Elastic Elements. *Zwischenbericht. ZB-86*, Technische Universität, Sttutgart..

13. Vlase, S.(1992) Finite Element Analysis of the Planar Mechanisms: Numerical Aspects. *Applied Mechanics* - 4. Elsevier, 90-100.

14. Vlase,S.(1999) The Liaison Forces Eliminating in the Elasto-Dynamic Analysis of Mechanical Interconnected Systems via Finite Element Method, *The Conference ARA*, Liege.

MULTIBODY MODEL OF HUMAN WALKING

M. WOJTYRA
Institute of Aeronautics and Applied Mechanics
Warsaw University of Technology,
Nowowiejska 24, 00-665 Warsaw, POLAND
E-mail: mwojtyra@meil.pw.edu.pl

1. Introduction

Researches into human gait have a wide range of applications in medicine, ergonomics, sport science and technology. Most often methods of multibody dynamics are used when investigation is focused on mechanical aspects of the gait [1], [2], [15], [17].
The inverse dynamics approach is commonly adopted in a human gait analysis. Displacements of the human body segments and ground reaction forces are known from measurements. The joint reaction forces and muscle net torques (which cannot be measured directly) are calculated. Since the ground reaction forces are known it is not necessary to model the foot-ground contact. The inverse dynamics approach requires measured displacements to be differentiated twice in order to obtain accelerations. The choice of filtering and smoothening methods affects significantly the obtained results [1], [16]. The errors in alignment of ground reaction force and foot affect the results as well [9].
In this paper a direct dynamics approach to the human gait analysis is presented. The direct dynamics approach is usually taken when a walking machine with its control system is investigated. It is seldom used to human gait simulation.
In the presented model the measurements of displacements of the human body segments are treated as a gait patterns (i.e. the patterns of relative motion in joints). The net torques (generated in the way that enables realisation of the gait patterns) are applied to a mechanical system and the direct problem of dynamics is solved. The foot-ground contact is modelled. For the gait stabilisation a simple closed loop control algorithm is introduced into the simulational model.
Though the method presented here is more complicated than the traditional one, there are some advantages. It is possible to predict system behaviour, whereas inverse dynamics approach is restrained to reconstruction only. Moreover, since accelerations are calculated, there is no need to differentiate measured displacements twice. Ground reaction forces as well as feet positions are calculated, so the alignment of foot position with reaction force is no longer a problem. And finally, the simulation is not limited by the number of measured gait cycles (usually one or two), since gait patterns can be extrapolated.

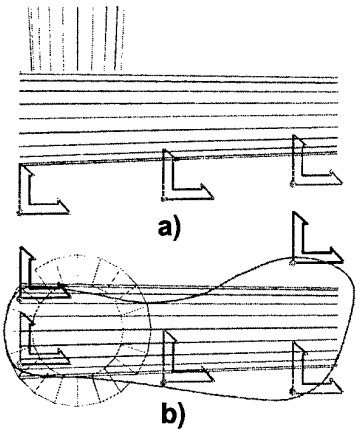


Figure 2. Ground reaction forces acting on foot.
a) side view, b) top view.

$$F_z = \begin{cases} MAX(0, -k\,d_z - c\,v_z) & d_z < 0 \\ 0 & d_z \geq 0 \end{cases}$$

where:
d_z - z coordinate of the point of force application,
v_z - z component of velocity ($v_z = \dot{d}_z$),
k - stiffness,
c - damping.
Damping c is a non-linear function of ground penetration p_z ($p_z = MAX[0, -d_z]$):

$$c = \begin{cases} c_{max}\left(\frac{3}{h^2}p_z^{\,2} - \frac{2}{h^3}p_z^{\,3}\right) & p_z < h \\ c_{max} & p_z \geq h \end{cases}$$

where: h, c_{max} - constant values.

A number of test simulations was performed in order to choose proper values of stiffness k of the foot in shoe and proper values of damping parameters h and c_{max}. The ground penetration p_z corresponds to the deflection of foot and shoe during walking. The maximum value of p_z observed during simulation was equal to 2 cm, which agrees well with experimental data [7]. The chosen level of damping is high enough to prevent the foot from "bouncing" after hitting the ground, which also agrees with experiments. The choice of parameters k, c_{max} and h is not crucial – it was proved that after the 20% change of the value the model behaviour has remained almost unchanged.

The tangent reaction force T is represented in terms of a pseudo-Coulomb friction model (in this model of friction there is no stiction, i.e. the bodies are moving relative each other at a negligibly small velocity). At the beginning velocity of sliding is calculated:

$$v_p = \sqrt{v_x^{\,2} + v_y^{\,2}}$$

where:
v_x - x component of velocity of sliding,
v_y - y component of velocity.
Then a modulus of friction force is calculated:

$$T = \mu'\, F_z$$

where: μ'- non-constant friction coefficient.
The dependence of μ' on v_p is given by:

$$\mu' = \mu\,\frac{2}{\pi}\,\mathrm{arctg}\,\frac{v_p}{\lambda}$$

where:
μ - Coulomb friction coefficient (constant value),
λ - constant value.
Finally, the tangent force components are calculated:

$$F_x = -T\frac{v_x}{v_p + \varepsilon}, \quad F_y = -T\frac{v_y}{v_p + \varepsilon}$$

where: ε - small constant value.

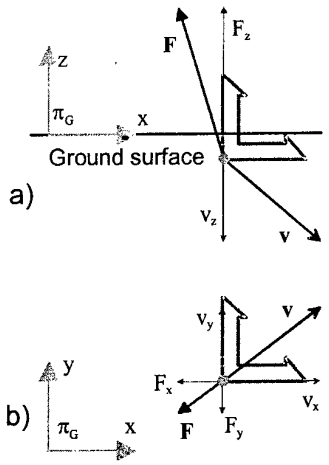


Figure 3.Force and velocity:
a) side view, b) top view.

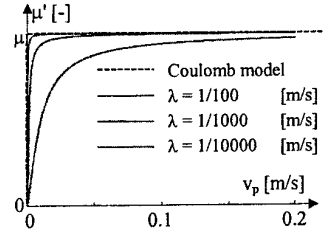


Figure 4. Friction coefficient dependency on the relative velocity.

2. Multibody Model

2.1. BACKGROUND

There exist various methods for modelling bipedal walking. Some multibody models for the gait simulation represent bipeds of a very simple (not anthropomorphic) kinematic structure [3]. Other models are limited to two dimensions [8], [11], [12], [18].
Walking is a special kind of the multibody system motion with the occurrence of impacts, friction and slipping. All these phenomena observed during the foot-ground contact have to be introduced into the model to make it realistic. In some multibody models of the biped the foot-ground contact is modelled by the additional kinematic joint [10], [12] or temporal fixing of the supporting foot to the ground [3]. Different models are used for single and double support phases. In this type of models the non-slip and non-impact conditions are assumed.
There is another group of models, in which the forces between foot and ground are modelled in a more realistic way [5], [6], [7], [18], [19]. Control of the model of this type is more difficult, since the control system must prevent the biped from slipping and the biped must absorb the shocks caused by impacts.
The model presented here is three-dimensional and reflects the kinematic structure and mass properties of the human locomotion apparatus. Both the slips and impacts are taken into consideration.

2.2. KINEMATICAL AND MASS PROPERTIES

The model and its kinematic scheme is presented in the Fig. 1. The model consists of 8 rigid parts. For the sake of simplicity, the trunk is modelled as two parts connected by revolute joint. The inertia properties of head and arms are included in trunk properties. Lower parts of human body are modelled more realistically. Each leg consists of 3 parts: thigh, shank and foot. Each hip joint is modelled as three consecutive revolute joints. These three joints are kinematically equivalent to a spherical joint. Each knee and each ankle joint is modelled as two consecutive revolute joints. The modelled biped has 21 degrees of freedom. To account for elasticity of human tissues, so-called wobbling masses are introduced to the model (in this case number of DOF increases to 46).

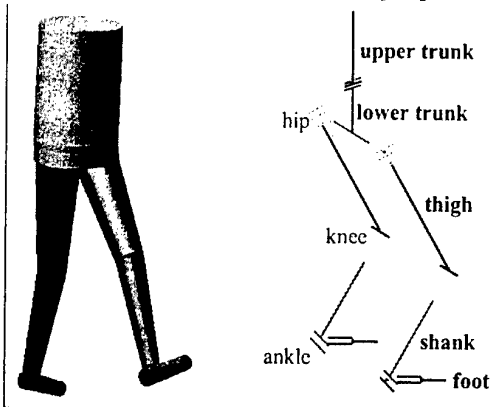


Figure 1. The general view and kinematic scheme.

2.3. GROUND REACTION FORCES

The impact and friction effects are considered in the ground reaction forces modelling. The ground is represented by a flat rigid surface. A set of 5 force vectors acting on each foot is used to model ground reaction forces. Fig. 2 presents points of force application. Normal to the ground (ground surface coincides with xy plane of global coordinate system) reaction force F_z is modelled using following function:

3. Control System

3.1. GAIT PATTERNS

Parameters characterising walk of the subject person were measured. A two-camera system was used to cinematographically measure trajectories of selected points of human body. The relative angles in each joint were calculated from the measurement results. The linear and angular parameters of absolute motion of the lower trunk were also calculated. The measured results were discrete. For the simulation purposes, however, a continuous and periodical function (periodical function can be easily extrapolated) is needed, which was obtained by approximating data by the appropriate Fourier series. These functions are called 'gait patterns'. During approximation process periodicity and symmetry of the gait were assumed.

3.2. DRIVING TORQUES

For the direct dynamics simulation all joints were equipped with actuators. The torque M generated by the actuator located in a joint was a function of: the current value of the joint angle φ and the desired joint angle value φ_0 given by the gait patterns:

$$M = \kappa(\varphi_0 - \varphi) + \upsilon(\dot{\varphi}_0 - \dot{\varphi})$$

where: κ, υ - constants.

This formulation of the torque equation (proposed earlier in [14]) ensures that the realised relative angle is close to the angle given by the gait pattern. The actuator can be treated as a motion generator (that strictly realises the gait pattern) connected in series with a spring-damper element. This spring-damper element is necessary to obtain a proper response to impacts (the system reaction to the impact is incorporated in the gait pattern, however heel strike occurs usually not exactly at the moment prescribed by the gait pattern - the system responses to the impact too early or too late).

Due to both the measurement inaccuracy and additional operations (making it symmetric and periodical) the gait pattern suffers from relatively big errors. When gait patterns for joints are precisely realised the absolute motion of the trunk is left uncontrolled. If the biped started to fall the control system would not react. The only way to control the absolute motion of the trunk is to apply a control algorithm, which instantaneously modifies the prescribed joint motions to prevent the biped from loosing its stability.

3.3. CONTROL ALGORITHMS

When the torques applied in joints depended only on current and desired value of the joint angle the relative positions of biped links with respect to each other were controlled, but the position of the whole biped with respect to the ground was not controlled. Therefore an additional external force and torque which supports the trunk was introduced.

The concept of external force was helpful in the process of control algorithm synthesis. The proposed control algorithm is a heuristic one. It consists in incorporating some feedback information, i.e. some quantities dependent on the current position and orientation of the biped trunk (with respect to the ground) into the function defining the torques in selected joints.

The method for the control algorithm construction will be detailed considering a simple algorithm for the trunk yaw angle stabilisation.

Firstly, the component of the supporting torque that corresponds to the yaw angle was set to zero. The remaining components of the supporting force and torque remained unchanged. Then the gait simulation was performed. Animation showed that after a few steps the biped turned about yaw axis and then fell over. It was not surprising, since the yaw angle was not controlled. The heuristic method proposed in this case consisted in introducing some changes into the torques that drive left and right hip joints (each hip joint consists of three consecutive revolute joints: α_1, α_2 and α_3). These changes were made in order to prevent the biped trunk from not controlled turning about the yaw axis. The functions defining torques in α_2 and α_3 joints were modified:

$$M_{\alpha_2} = \kappa\,(\alpha_{02} - \alpha_2 + \sin\alpha_1 \Delta_\xi) + \upsilon\,(\dot{\alpha}_{02} - \dot{\alpha}_2)$$

$$M_{\alpha_3} = \kappa\,(\alpha_{03} - \alpha_3 - \cos\alpha_1 \Delta_\xi) + \upsilon\,(\dot{\alpha}_{03} - \dot{\alpha}_3)$$

$$\Delta_\xi = \begin{cases} \delta\,(\theta_0 - \theta) & \text{support phase} \\ 0 & \text{swing phase} \end{cases}$$

where: θ_0, θ - yaw angle given by the gait pattern and the current yaw angle, respectively
In the next few simulations of the biped gait the appropriate (i.e. leading to a stable gait) value of δ constant was chosen.
Similar procedures (the yaw angle stabilisation algorithm is one of the simplest) were used for the other components of the supporting force and torque. As a result a stable gait without the additional external supporting force was achieved. The stable walking is a result of cooperation of several algorithms. The movement realised by the biped is slightly different from that given by the gait patterns. These differences are necessary for stabilisation of the gait. The utilised control algorithms are rather simple, nevertheless they enable the biped to walk.

4. Simulation Results

During simulations the integration procedures were changed to ensure that simulation results remained unchanged. The multibody model behaviour sensitivity to the control algorithm and the model parameters was checked. It was shown that positioning of the three-component force objects on the feet exerts the strongest influence on the model behaviour.
In the direct dynamics approach the ground reaction forces are computed (they do not play the role of input data). Comparison of the measured ground reaction forces with the calculated ones was used to validate the model. The comparison is illustrated in Fig. 5a. The calculated and the measured results are similar, however there are some differences. These differences are caused mostly by the fact that foot is modelled as one rigid body and a contact between foot and ground is reduced to five points only. To obtain better results the model of foot and

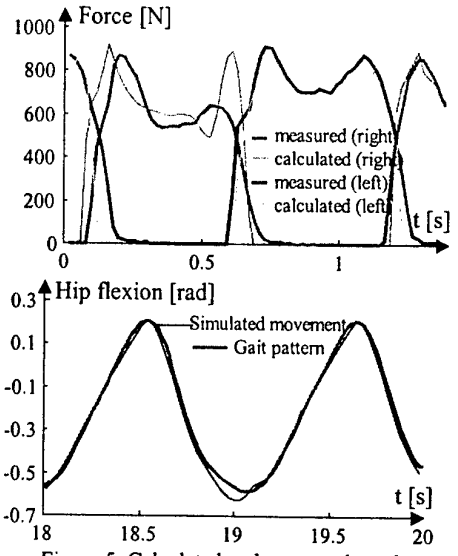


Figure 5. Calculated and measured results:
a) vertical ground reaction forces,
b) right hip flexion angle.

the model of contact should be made more sophisticated. Approximations of measured walk parameters by periodical and symmetric gait patterns is another reason for differences - simulated ground reaction forces are much more regular, than measured ones. Simplifications made during the modelling process are also factors in obtained differences.

The observed differences between measured and simulated data are less than 15% of maximal values, which is a decent result when biomechanical calculations are considered.

The control algorithms introduce some differences between the simulated motion and prescribed gait patterns (see Fig. 5b). These differences are relatively small, however big enough to maintain the biped stability.

5. References

1. Allard P., Stokes I. A. F., Blanchi J.-P. (ed.): *Three-Dimensional Analysis of Human Movement.* Human Kinetics Publishers, Champaign, Illinois, 1995.
2. Berme N., Cappozzo A. (ed.). *Biomechanics of Human Movement.* Ohio, USA, 1990.
3. Blajer W., Schiehlen W. O. *A control scheme for biped walking without impacts.* 9th CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators, pp. 313-321, Springer-Verlag 1993.
4. Bourassa P., Prince F., Meier M., Roy Y.: *A simple walking model; simulation and animation.* Proc. of VIth International Symposium on Computer Simulation in Biomechanics, pp. 1 - 4, Tokyo, 1997.
5. Gerristen K. G. M., Van den Bogert A. J., Nigg B. M.: *Direct dynamics simulation of the impact phase in heel-toe running.* Journal of Biomechanics, Vol. 28, pp. 661 - 668, 1995.
6. Gilchrist L. A., Winter D. A.: *A two-part, viscoelastic foot model for use in gait simulations.* Journal of Biomechanics, Vol. 29, pp. 795 - 798, 1996.
7. Güler H. C., Berme N., Simon S. R.: *A viscoelastic sphere model for the representation of plantar soft tissue during simulations.* Journal of Biomechanics, Vol. 31, pp. 847 - 853, 1998.
8. Koopman B., Grootenboer H. J., de Jongh H.: *An inverse dynamics model for the analysis, reconstruction and prediction of bipedal walking.* Journal of Biomechanics, Vol. 28, pp. 1369 - 1376, 1995.
9. McCaw S. T., DeVita P.: *Errors in alignment of center of pressure and foot coordinates affect predicted lower extremity forces.* Journal of Biomechanics, Vol. 28, pp. 985 - 988, 1995.
10. Onyshko S., Winter D. A.: *A mathematical model for the dynamics of human locomotion.* Journal of Biomechanics, Vol. 13, pp. 361 - 368, 1980.
11. Ouezdou F. B., Bruneau O., Guinot J.-C.: *Dynamic analysis tool for legged robots.* Multibody System Dynamics, Vol. 2, No 1, pp. 369 - 391, 1998.
12. Pandy M. G., Berme N.: *Synthesis of human walking: a planar model for single support.* Journal of Biomechanics, Vol. 21, pp. 1053 - 1060, 1988.
13. Seireg A., Arvikar R.: *Biomechanical Analysis of the Musculoskeletal Structure for Medicine and Sport.* Hempshire Publishing Co, New York, 1989.
14. Van den Bogert A. J., Schamhardt H. C., Crowe A. *Simulation of quadrupedal locomotion using a rigid body model.* Journal of Biomechanics, Vol. 22, ss. 33 - 41, 1989.
15. Vaughan C. L., Davis B. L., O'Connor J.: *Dynamics of Human Gait.* Human Kinetics Publishers, Champaign, Illinois, 1992.
16. Vint P. F., Hinrichs R. N.: *Endpoint error in smoothing and differentiating raw kinematic data: an evaluation of four popular methods.* Journal of Biomechanics, Vol. 29, pp. 1637 - 1642, 1996.
17. Winter D. A. *Biomechanics of Human Movement.* John Wiley & Sons Inc., 1979.
18. Wojtyra M.: *Applicability of human gait patterns to biped gait control - computer simulation.* Proc. of the 12th CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators, pp. 393 - 400, CISM, Udine, 1998.
19. Wojtyra M.: *Dynamical simulation of human walking.* Proc. of Tenth World Congress on the Theory of Machines and Mechanisms, pp. 1853 - 1858, Oulu, Finland, 1999.

SOME RESULTS OF WHEEL-RAIL CONTACT MODELING*

V. N. YAZYKOV
Bryansk State Technical University
Bulv. im. 50-letiya Oktyabrya, 7, 241035, Bryansk, Russia

1. Introduction

Research of railway vehicle dynamics by means of mathematical models is the necessary stage for providing the vehicles with improved characteristics. Many problems such as dynamic stability, computation of wheel wear and others can be successfully solved by using a computer-aided multibody model of the vehicle. Significant part of the model is the description of forces at contact between wheel and rail. Computation of these forces is one of the most CPU time-consuming operations during the simulation process. Mathematical models of the contact forces often lead to stiff equations of motion because of high contact stiffness. In this paper an approximate non-stiff method for computing the non-elliptical contact problem and some results of its implementation are presented.

2. Model of rail-wheel contact forces

2.1. APPROXIMATE SOLUTION OF THE NORMAL PROBLEM

The simplest way to solve normal contact problem is to replace the real shapes of bodies at contact with quadratic surfaces and then use the Hertzian solution. Though this method is very fast, actually the contact patch is often far from elliptical and it is necessary to use more exact methods than the Hertzian solution (e.g. for conformal contact or computation of evolution of a wheel profile due to wear). The corresponding algorithm must be fast to be successfully used in simulation of multibody system dynamics.

The method by Kik and Piotrowski [1] for calculation of normal load and distribution of normal pressure is quite fast. But in our opinion the contact force model, in which the forces depend on interpenetration of contacting bodies, leads to stiff equations of motion. We modified the method by Kik and Piotrowski to decrease the stiffness of the equations. The parabolic distribution of the normal pressure in the direction of rolling instead of elliptic one was also used to decrease calculation efforts.

* Supported by the Russian Foundation for Basic Research under the grants 02 -01-00364 and by the scientific program "Universities of Russia – Basic Research" (UR.04.01.046).

According to paper [1] the approximate contact patch is defined in the following way. The wheel and the rail are considered as a body of revolution and a cylindrical surface, respectively. The surfaces are interpenetrated in a depth δ as rigid bodies, Figure 1. The function $u(x,y)$ specifies the interpenetration of the surfaces at the point (x,y). It satisfies the equation

$$u(x,y) = \delta - z(x,y), \; z(x,y) = \frac{x^2}{2R} + h(y), \tag{1}$$

where $z(x,y)$ is the function, which specifies the distance between the points for $\delta = 0$, R is the wheel radius at the contact point.

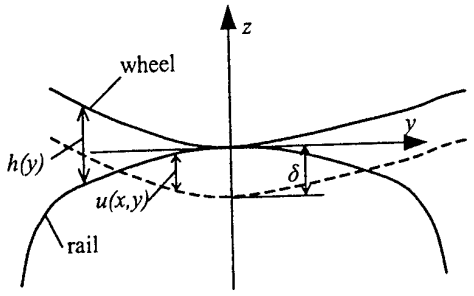


Figure 1. Wheel-rail contact.

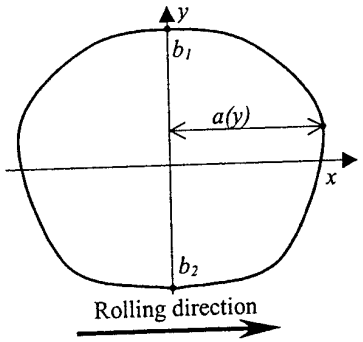


Figure 2. Contact patch.

Edge of an approximate contact patch is determined as a line of intersection of the surfaces, Figure 2.

The dependence of the intersection line on the lateral coordinate is

$$a(y,\delta) = \sqrt{2R(\delta - h(y))}, \tag{2}$$

The roots b_i of the equation

$$\delta = h(y) \tag{3}$$

determine the length of the patch along the lateral axes.

The number of separate zones of contact is equal to half of the number of roots of equation (3).

The materials of the wheel and the rail are considered to be identical. Assuming that the bodies at contact are half-spaces, the value of δ can be estimated. The deflection at point $(0,0)$ can be found with the help of the Boussinesq's influence function as

$$\omega(0,0) = \frac{1-\nu^2}{\pi E} \iint_C \frac{p(x,y)}{\sqrt{x^2+y^2}} dxdy, \tag{4}$$

where $p(x,y)$ is the distribution of normal pressure. According to the assumption about materials, the wheel and rail deflections at contacting points are equal, so $\delta = 2\omega(0,0)$. But in the reality the bodies at contact cannot interpenetrate and deflections occur, so the interpenetration region enclosures the contact patch if the influence function is uni-directional. Granting this fact the bodies are interpenetrated in depth $\delta_0 < \delta$ (Kik and Piotrowski recommend to take the value of δ_0 equal to 0.55δ for the elliptic distribu-

tion in the direction of rolling).

We used the assumption about proportionality between the normal pressure $p(x,y)$ and function of interpenetration $u(x,y)$ instead of the half-elliptic distribution of the pressure in the rolling direction [1]. It leads to decreasing the calculation efforts by reducing the double integrals to single ones, though such distribution is less accurate.

So the distribution of the normal pressure is

$$p(x,y) = k_p u(x,y)\,, \tag{5}$$

where k_p is a proportionality factor.

It is supposed, that the normal force N at the contact point is available from solution of the dynamic equations. The interpenetration of the bodies is used only to calculate the normal pressure distribution and contact patch and neglected in the dynamic equations. Thus, the normal force depends on the vertical and lateral stiffness of the rail-track system and this model is not stiff.

Using (1) – (5) the interpenetration in the case of single zone in the contact patch is

$$\delta = 2\omega(0,0) = 2\frac{1-\nu^2}{\pi E} k_p \int_{b_1}^{b_2}\int_{-a}^{a} \frac{\delta - \dfrac{x^2}{2R} - h(y)}{\sqrt{x^2+y^2}}\,dxdy = \tag{6}$$

$$2\frac{1-\nu^2}{\pi E} k_p \int_{b_1}^{b_2}\left[\left(\delta - h(y) + \frac{y^2}{4R}\right)\cdot \ln\left(\frac{a+\sqrt{a^2+y^2}}{|y|}\right) - \frac{a}{4R}\sqrt{a^2+y^2}\right]dy.$$

The normal contact force is calculated as

$$N = \iint_C p(x,y)\,dxdy = \iint_C k_p u(x,y)\,dxdy = k_p 2\int_{b_1}^{b_2}\left(\delta a - h(y)a - \frac{a^3}{6R}\right)dy\,. \tag{7}$$

Using (6), (7) we obtain the following nonlinear equation:

$$N = \frac{\delta\pi E}{2(1-\nu^2)}\cdot \frac{\displaystyle\int_{b_1}^{b_2}\left(\delta a - h(y)a - \frac{a}{6R}^3\right)dy}{\displaystyle\int_{b_1}^{b_2}\left[\left(\delta - h(y) + \frac{y}{4R}^2\right)\cdot \ln\left(\frac{a+\sqrt{a^2+y^2}}{|y|}\right) - \frac{a}{4R}\sqrt{a^2+y^2}\right]dy} \tag{8}$$

The solution of (8) is the interpenetration δ. Taking into account that $\delta_0 < \delta$, we found the approximate contact patch and distribution of normal pressure. Though the process of solution is iterative, the convergence of solution requires small number of iterations if start δ values are taken from the previous integration step. The method was tested on computer-aided model of the railcar AS-4, which was realized in the program package "Universal Mechanism". The interpenetrations and the number of iterations necessary for solving the normal contact problem are represented in Figure 3. The mean number of iterations required for the vehicle simulation in an even curve is 1.71, in an uneven curve 2.63.

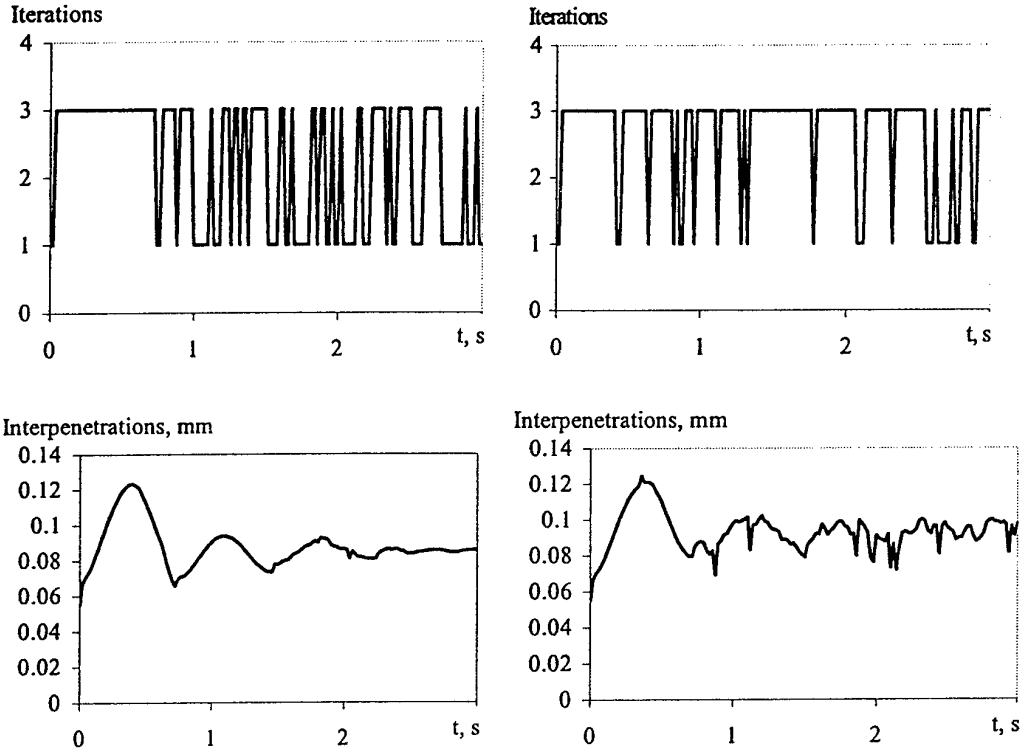


Figure 3. Simulation results (left – even curve, right – uneven curve).

2.2. CALCULATION OF TANGENTIAL FORCES

The most of the tangential force models (so-called creep forces in railway dynamics) are based on the assumption that these forces depend on the creepage (ξ_x is the longitudinal creepage, ξ_y is the lateral creepage) and the spin φ:

$$\xi_x = \frac{V_x^w}{V}, \quad \xi_y = \frac{V_y^w}{V}, \quad \varphi = \frac{\omega_n^w}{V},$$

where V_x^w, V_y^w are the corresponding projections of velocities of wheel as a rigid body at contact point, ω_n^w is the projection of wheel rotational velocity onto normal to the tangential plane, V is the vehicle velocity.

We used the FASTSIM algorithm [2, 3] to calculate the creep forces at the contact. The tangential pressure was found from system of equations

$$\begin{cases} \partial q_x / \partial x = (\xi_x - \phi y)/L \\ \partial q_y / \partial x = (\xi_y + \phi x)/L, \end{cases}$$

where q_x, q_y – the components of the tangential pressure, L – flexibility.

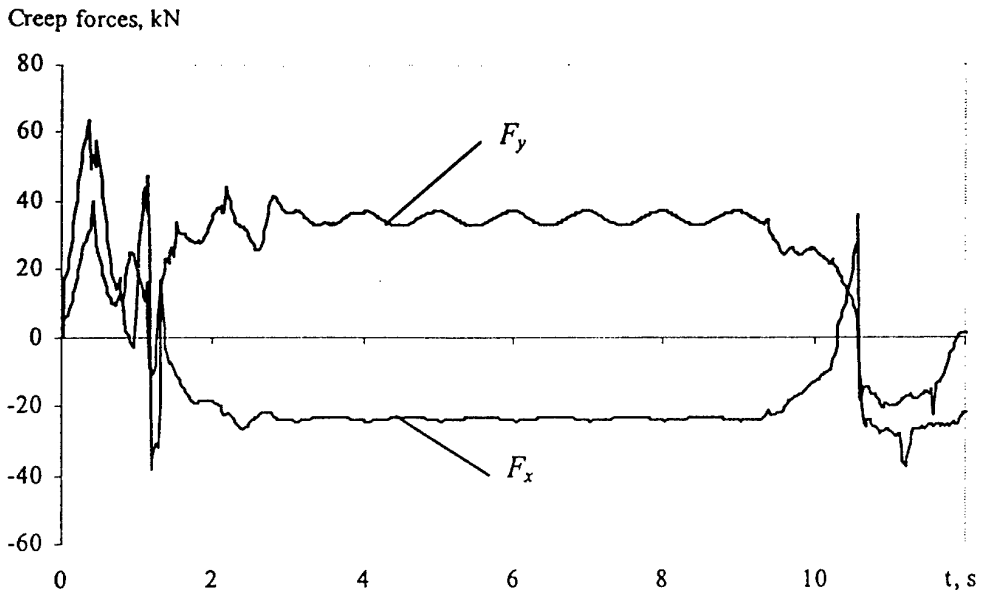


Figure 4. Simulation results for even curve.

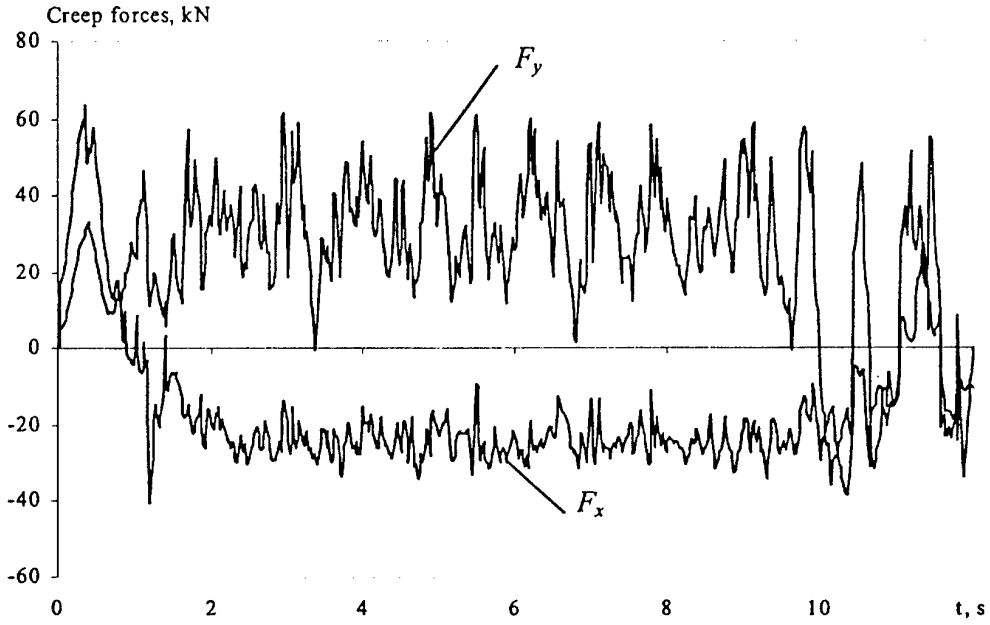


Figure 5. Simulation results for uneven curve.

The obtained values of pressure q are tested for satisfying the Coulomb's law. The contact patch and the distribution of normal pressure were computed by using the method described in Section 2.1.

To determine the value of flexibility L we calculated an equivalent ellipse such that the area of non-elliptic contact patch is equal to the area of the ellipse [4]. The semi-axis of the ellipse in the rolling direction a is set equal to the maximal half-length of the

non-elliptical patch. The lateral semi-axis of the equivalent ellipse can be found from the equation

$$b = \frac{A_{ne}}{\pi a},$$

where A_{ne} is the area of the non-elliptical contact patch.

Figures 4, 5 show calculated creep forces for the simulation of tested vehicle in a curve.

3. References

1. Kik, W. and Piotrowski, J. (1996) A fast approximate method to calculate normal load at contact between wheel and rail and creep forces during rolling, *Proceedings of 2nd mini. conf. Contact Mechanics and Wear of Rail/Wheel Systems*, 52-61.
2. Kalker, J.J. and Piotrowski, J. (1989) Some new results in rolling contact, *Vehicle System Dynamics* **18**, 223-242.
3. Kalker, J.J. (1973) Simplified theory of rolling contact, *Delft progress report, Series C: Mechanical and aeronautical engineering and shipbuilding* **1**, 1-10.
4. Harder, R.F., Meekisho, L.L., Jones, J. and Rhoades, V. (1996) Generalized approximation of wheel-rail creep forces and contact patch frictional work using neural network simulation, *Proceedings of 2nd mini. conf. Contact Mechanics and Wear of Rail/Wheel Systems*, 21-33.

MODIFICATION OF ROAD PROFILE TO COMPENSATE TIRE NONLINEARITY IN LINEAR TIRE MODEL

Wan-Suk Yoo
Professor, Mechanical Engineering, Pusan National University
Kumjung-Ku, Pusan, Korea 609-735
wsyoo@pusan.ac.kr Tel 82-51-5102328, Fax 82-51-5129835

Myung-Gyu Kim
Deputy general manager, Hyundai motor company, Ulsan plant, Ulsan, Korea

Kwang-Suk Kim
Assistant Professor, Automotive Engineering, Inha Technical College, Inchon, Korea

Abstract

For the stress and durability analysis of vehicle components with computer simulation, it is necessary to find forces acting on the vehicle components due to the road profile undulation. In this study, road profiles are regenerated to preserve the same PSD of wheel responses with a linear tire model. The frequency response function between road and wheel, the digital signal processing method, and DADS program are used. Simulation results of load transfer at suspension components using this virtual road profiles are presented.

1. Introduction

For the analysis of stress and vibration of vehicle components moving on the road, experimental or analytical method can be used. In the experimental method, Belgian road or cross country road are commonly used as road inputs for durability test. Since the experimental methods generally require high cost and considerable time, computer simulation method is usually employed for the analysis of reaction forces, vibrations, and durability. In the computer simulation, however, it is difficult to assign load conditions and boundary conditions properly. Boundary load conditions for the stress or vibration analysis are sometimes assumed as the forces on the wheel during braking or bumping.
Liu and Haug[1,2] used the computer simulation methods for fatigue life estimation of machine components. Baek[3,4] suggested the dynamic load history, which is calculated by flexible multibody dynamic analysis, for fatigue analysis. In previous researches on road profiles, most of their results were focused on the statistical road profile[5,6,7]. Thus, the wheel response due to the road input was different from actual response. If it is possible to

calculate the same PSD and time signal responses as experimental results, force boundary conditions for the analysis could be accurately imposed. In this paper, a new method is suggested to match the load conditions at the suspension by regenerating road profiles. The tire model and road profile are modified preserving the same PSD of wheel movements as experimental ones by the digital signal processing technique.

2. Process for Road Profile Regeneration

If it is assumed to be a linear system between road and tire, road profiles could be reproduced by PSD(Power Spectral Density) of wheel acceleration and FRF(Frequency Response Function) between road input and wheel acceleration. A vehicle is modeled as a MIMO (multi input multi output) system, in which the outputs represent acceleration, velocity and displacement of the vehicle due to the road input. PSD value from the experiment is assigned as an objective function, and the PSD from computer simulation is modified to be matched the experimental one.

2.1. GENERAL PROCEDURE

The general procedures for road profile regeneration are;

1) Measure the acceleration PSD of the wheel center from experiment.
2) Calculate FRF of the vehicle model from computer simulation.
3) Define PSD of the initial road profile for dynamic simulation.
4) Carry out dynamic simulation with the initial road profile.
5) Compare PSDs from experiment and computer simulation, and define an error function.
6) If the magnitude of the error is small enough to accept, then stop.
7) If not, create a modified road profile by IFRF & IFFT.
8) Carry out dynamic simulation with the modified road profile, and return to step 4.

2.2. EXPLANATION OF EACH PROCESS

2.1.1. *Measure acceleration PSD of the wheel center*

For a precise modeling, suspension stiffness and geometry data in quasi-static condition are usually measured from the suspension parameter measuring device(SPMD). The acceleration PSDs of the wheel center are also measured from the experiment, and are used as an objective function to be matched in the computer simulation.

2.1.2. *FRF between road profile and wheel acceleration*

For the computer simulation, a computational vehicle model is developed with the DADS program in this research. Model validation in dynamic response can be done by comparison between test results and simulation ones. Frequency range for model validation is set as 20Hz

in this research, which is usual in the validation of a suspension. To obtain FRF between road profile input and wheel acceleration, a white noise signal is applied as a road input.

$$X_i(f_k) = \Delta t\, X_{ik} = \Delta t \sum_{n=0}^{N-1} x_{in} \exp[\frac{-j2\pi k n}{N}] \quad (1)$$

Auto spectral density(ASD) of road profile, cross-spectral density(CSD) between road profile and wheel response, and a FRF matrix between wheel and road are calculated as the following equations.

ASD
$$G_{xx}(f_k) = \frac{2}{n_d N \Delta t} \sum_{i=1}^{n_d} |X_i(f_k)|^2 \quad , k = 0,1 \cdots \frac{N}{2} \quad (2)$$

CSD
$$G_{xy}(f_k) = \frac{2}{N \Delta t} [X^*(f_k)\, Y(f_k)] \quad , k = 0,1 \cdots \frac{N}{2} \quad (3)$$

FRF
$$H(f) = \frac{G_{xy}(f)}{G_{xx}(f)} = |H(f)|\, e^{-j\phi(f)} \quad (4)$$

where n_d and $\phi(f)$ are data sampling frame and phase of FRF, respectively.

Since the interesting frequency is within 20Hz, the sampling frequency is set as 100Hz to avoid numerical errors at the boundary. The 1024 data are sampled in each frame. *Figure 1* and *Figure 2* shows the auto-spectral density of initial road profile and wheel acceleration, respectively. *Figure 3* shows cross-spectral density of the wheel acceleration and road input.

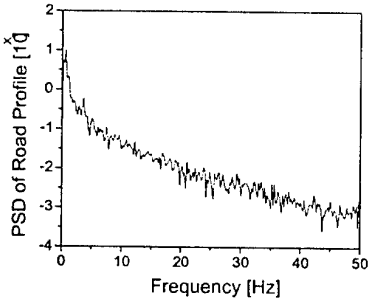


Figure 1. Auto-spectral density of initial road profile

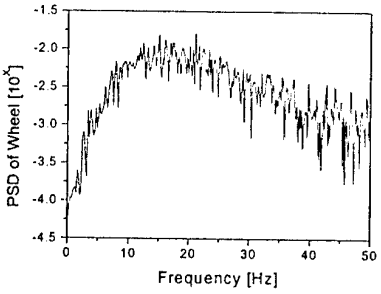


Figure 2. Auto-spectral density of wheel acceleration

In a passenger car, 4-wheel road inputs may generate a 4x4 FRF matrix. If a suspension system is independent type, the diagonal terms of FRF matrix are dominant. Then, the FRF can be obtained from one suspension. *Figure 4* shows the frequency response function of road input and wheel acceleration response.

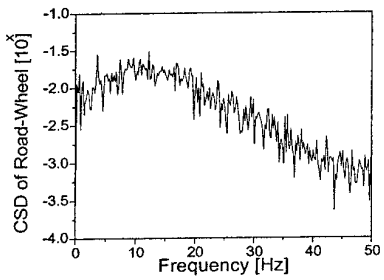


Figure 3. Cross-spectral density of wheel acceleration and road input

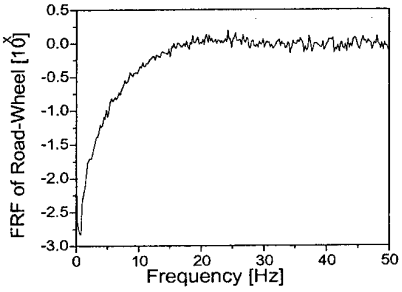


Figure 4. Frequency response function of road input and wheel acceleration

2.1.3. *Initial Road Profile*

To make the first iteration in computer simulation, PSD of the initial road profile should be imposed. In this paper, the initial road profile is assumed to be a white noise as following;

PSD of initial road profile: $G_z(f) = \frac{aV}{f^N}$ (5)

where a, V, f, and n are road roughness, vehicle velocity, frequency and exponent, respectively. Using this road profile PSD, time signal of the initial road profile is made for dynamic analysis. *Figure 4* shows the auto-spectral density of the initial road profile.

2.1.4. *Dynamic simulation with initial road profile*

Dynamic analysis is carried out with the initial road profile using DADS program. From the dynamic simulation result, wheel acceleration data are obtained and processed to find the PSD.

2.1.5. *Error Function*

Comparing the computed PSD with the measured one, error functions are defined at each frequency.

Error function: E(f)= PSDtest – PSDsimulation (6)

2.1.6. *Regeneration of Road Profile*

Correct response function is defined by the error function and response function of previous iteration step.

Corrected response: $P(f)_i = PSD_{i-1} + E(f)_i \cdot GAIN$ (7)

The wheel accelerations within the prescribed frequency range are measured at the same time. Value of the GAIN is set between 0.5 and 1.0 to avoid overdriving and divergence of wheel response. Using the corrected function and the FRF of wheel-road input, make the next modified driving file as;

Modified road profile : $D(f)_i = H^{-1}(f) \cdot P(f)_i + D(f)_{i-1}$ (8)

Although many nonlinear components are included in a suspension system, vertical stiffness of tire is almost constant when the tire is in contact with the ground. Therefore, tire and road profile system can be assumed as linear.

2.1.7. *Convergence of error function*

At each iteration step, the modified road input driving file is linked to DADS suspension model. Performing a data processing, the PSD of wheel response is calculated and compared with the desired PSD. For the convergence of error function, create a modified road profile D(f) and analyze the suspension model repeatedly until the error function is converged within the desired range. *Figure 5* and *Figure 6* show the road profile and vertical acceleration of wheel after the first iteration. These graphs show that there are large deviations from the original road profile.

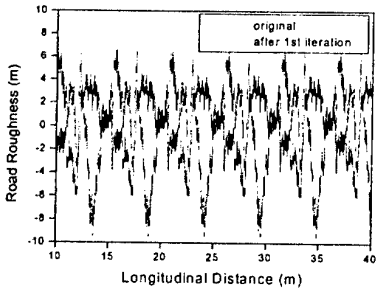


Figure 5. Road profile after the first iteration

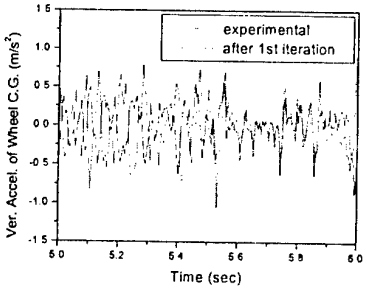


Figure 6. Vertical acceleration of wheel after the first iteration

After the second iteration, the deviations in the road profile and vertical acceleration are much decreased as shown in *Figure 7* and *Figure 8*.

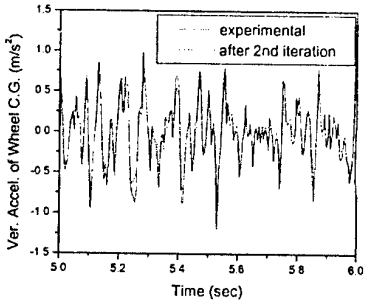


Figure 7. Road profile after the second iteration

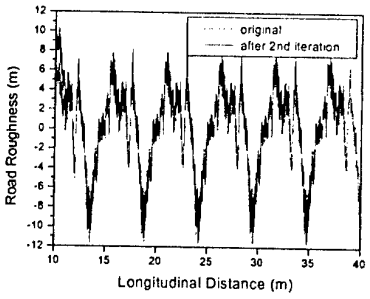


Figure 8. Vertical acceleration of wheel after the second iteration

3. Conclusions

1) A road profile regeneration method is proposed to preserve the same PSD level at the wheel center with a linear tire model. The proposed road profile technique is linked to a full vehicle model to predict the joint reaction forces. These results increase the reliability of

stress analysis for durability analysis.
2) Even though this research uses 1-channel to calculate the vertical directional motion, it can extend to 4-channel or 12-channel with the same idea.

Acknowledgments

This research was supported by the NRL(National Research Laboratory) project by the Korean government.

References

1. T.Liu, (1986) Ph.D. Thesis, *Computational Methods for Life Prediction of mechanical Components of Dynamic System*, The Univ. of Iowa, Iowa City, Iowa
2. E.J.Haug, T.S.liu, L.Johnson, W.K.Baek, et. Al (1986) *Progress in Flexible system Dynamics and Vehicle Life Prediction*, The Univ. of Iowa ,Technical Report 86-20
3. W.K.Baek, R.I.Stephens, and B.Dopker (1993) *Integrated Durability Analysis*, Trans. of the ASME, Journal of Engineerinf for Industry, Vol.115, No.4, pp.492-499
4. W.K.Baek (1990) Ph.D. Thesis, *Computational Life Prediction Methodology for Mechanical Systems Using Dynamic Simulation, Finite Element Analysis, and Fatigue Life Prediction Methods*, , The Univ. of Iowa, Iowa City, Iowa
5. C.J.Dodds, J.D.Robinson (1973) *The Description of Road Surface Roughness*, J. of Sound and Vibration, Vol. 31, No. 2, pp.175~183
6. K.M.A.Kamash, J.D.Robinson (1978) *The Application of Isotropy in Road Surface Modeling*, J. of Sound and Vibration, Vol. 57, No. 1, pp.89~100
7. A.N.Heath (1989) *Modeling and Simulation of Road Roughness*, Proceedings of 11th IAVSD Symp. The Dynamics of Vehicles on Roads and on Tracks, pp.275-284, August 21-25
8. J.G.Wendenborn (1966) *The Irregularities of Farm Field Roads and Fields as Sources of Farm Vehicle Vibration*, J. of Terramechanics, Vol. 3, No. 3, pp.9~40
9. A.N.Heath (1987) *Application of the Isotropic Road Roughness Assumption*, J. of Sound and Vibration, Vol.115, No.1, pp.131-144
10. ISO/TC108/SC2/WG4 N57 (1982) *Reporting Vehicle Road Surface Irregularities*
11. Y.M.,Pevzner and A.A.Tikhonov (1964) *Spatial Description of the Micro-Profile of Automotive Roads*, Automobil'naya Promyshienmost, Vol.30, No.1, pp.9-14
12. J.S.Bendat and A.G.Piersol (1986) *Random Data : Analysis and Measurement Procedures*, John Wiley & Sons, New York

MECHANICAL AND MATHEMATICAL MODELLING AND COMPUTER SIMULATION OF VIBRATION AND IMPACT PROCESSES IN THE 'MAN AND SHOOTING DEVICE' SYSTEMS

Applications in the archery and other relevant sports and physical exercises

I. ZANEVSKYY
Lviv State Institute of Physical Culture
Kostyushko str. 11, Lviv 79000, Ukraine

Abstract
Theoretical investigation of natural modes and frequencies of bow and arrow vibration has been done as a study of a boundary problem for the differential equation of the fourth order. The solution of the equation has been found in a form of polynomial series using a method of successive approximations. Data results for the first four natural frequencies and modes have been obtained and practical conclusions have been drawn. An archer's paradox and the spine phenomenon have been explained using the results of mathematical modelling. As a result of modelling and computer simulations an engineering method for matching bow and arrow parameters has been proposed. Comparative results for the wrong and right combination of these parameter values for the modern sport bow and two arrows have been presented.

1.Introduction

The problems of mechanical and mathematical modelling of the 'Man and device (or machine)' systems are exceptionally significant for the work environments in different fields of industry, agriculture, construction, transport, medicine (orthopaedics, prosthetics, medical engineering), and sports, for example, skiing, throwing, shootings, different playing with a ball etc.).
The sports played with mechanical devices represent a big proportion of the human competitive and recreational activity. However, the level of our knowledge and understanding of technique in these activities seems to lag behind those of other popular activities. There are some somewhat obvious reasons for this discrepancy. One of them is a non-sufficiently level of mechanical and mathematical modelling and computer simulation in this field.
A sport of the archery is a good model for a study the mechanical processes in the 'Man and device' system. Archery is a sport for people of all ages, and is challenging whether the individual is alone or with a group of people, since the competition is between the archer and his or her device (bow and arrows). Archery is a mass participation recreational sport such as in modern equipment is both necessary and desirable.
The purpose of the study is to develop the methods for quantitative and operative valid performance of the mechanical processes in 'Bow and archer' system.
The investigation of the archer's paradox was found by using high speed spark photography, which was undertook to secure direct evidence of what an arrow does as it leaves the bow [2]. The archer's paradox is the phenomenon that an arrow does not fly to its mark along the line represented by its axis. The forces acting on the arrow during its release do not quite coincide

with this axis, because the string force acts on the arrow in the bow plane. In the starting position the arrow does not lie in this plane, as its axis makes an angle of a few degrees with it. Even in the case that the nock and head points of the arrow do lie in the bow plane, the longitudinal arrow axis of the string force line does not quite coincide with the bow plane because the initial shape of the arrow axis is not quite straight. So, the string force line does not coincide with the line of a cross-section centre of the arrow. The released string pushes the arrow's nock point in the bow plane. Therefore the arrow will move forwards and thereby turn, slightly decreasing the angle with this plane. The impulse normal to the axis of the arrow caused by the release of the fingers from the string, as well as the column-like force of the string, pushes the arrow during its acceleration motion. These results in significant bending of the arrow shaft as it transits the bow. All of these factors allow the arrow to oscillate around the bow handle and to follow a straight course towards its target without striking the bow handle (Fig. 1). The arrow moves in the x-direction. The x-y plane is not horizontal but makes a slope with an angle of a few degrees for target archery and about forty-five degrees for flight shooting.

2.Mathematical Model

Potential energy V and kinetic energy T of a transverse motion system perpendicular to the bow flatness lateral plane xOy (Fig.2) are:

$$V = \frac{1}{2}\int_0^l EJ(y'')^2\,dx - \frac{a}{2}\int_0^l\left[\rho A\int_0^x (y'-2u')y'd\chi + m_1(y'-2u')y'\right]dx +$$

$$+\frac{1}{2}c\left(y_2-y_0\right)^2,\quad T = \frac{1}{2}m_0\dot{y}_0^2 + \frac{1}{2}\int_0^l \rho A\dot{y}^2dx + \frac{1}{2}m_1\dot{y}_1^2 + \frac{1}{2}m_2\dot{y}_2^2, \qquad (1)$$

where u is the deflection caused by the initial curvature of the arrow, y is the total deflection of an arrow shaft, y_0 is the deflection of an arrow tail, y_1 is the deflection of an arrow head, y_2 is the transverse displacement of the bow virtual mass, E is Young's modulus of the arrow shaft material, J is moment of inertia of an arrow shaft cross-section area, ρ is mass of a unit volume of a shaft material, l is the length of an arrow, A is the shaft cross-sectional area, m_0 is the mass of the arrow tail with virtual mass of a string, m_1 is the mass of the arrow head, m_2 is the virtual mass of the bow limbs, c is the virtual stiffness of the bow limbs and the string, a is the longitudinal acceleration of the arrow motion, (˙) and (′) are derivatives with respect to time t and the longitudinal co-ordinate x respectively. For a composite metal and carbon arrow shaft EJ is the arrow shaft bending stiffness. The virtual mass of string is equal to 1/3 of the whole mass of the string as the arrow leaves it [4].

Using Hamilton's principle for the mechanical arrow-bow system $\delta\int_{t_1}^{t_2}(T-V)dt = 0$, after substituting the expressions of T and V from (1) in the last equation we get the differential equation: $\left(EJy''\right)'' - \left(FY'\right)' + \rho A\,\ddot{y} = 0$, and boundary conditions:

$$x=0\quad y''=0,\quad m_2\ddot{y}_2 - c(y-y_2)=0,\quad \left(EJy''\right)' + (m+m_1)aY' + m_0\ddot{y} + c(y-y_2)=0;$$

$$x=l\quad y''=0,\quad \left(EJy''\right)' + m_1(aY'-\ddot{y})=0,$$

and initial conditions at $t=0\quad y=0,\quad y_2=0,\quad \dot{y}=0,\quad \dot{y}_2=0,$ (2)

where $m = \int_0^l \rho A dx$ is the arrow shaft mass, $Y = y + u$ is the total arrow shaft axis deflection, $F = -a(m_1 + \int_x^l \rho A d\chi)$ is longitudinal compress force in the arrow shaft body.

The arrow's constrained lateral vibration can be rated in order of a reduced problem of the system with concentrated parameters. We can then introduce the second order Lagrange equations. Assuming arrow acceleration at a certain moment is constant, we can write for a cylindrical arrow: EJ=const, ρF=const. The problem may be solved with the expressions in separated natural modes and time functions [3]:

$$y = \sum_k X_k(x) T_k(t), \qquad y_2 = \sum_k L_k T_k(t). \qquad (3)$$

where $X_k(x)$ are natural modes (eigenfunctions), $T_k(t)$ are time functions, L_k are bow handle amplitudes.

The solution of a space-time problem (2) has been reduced to the simultaneous boundary problem with independent natural modes. Substituting (3) in (2) and making necessary transformations we get differential equations for natural modes in a dimensionless form:

$$X_k^{IV} + \phi[(1 + \mu_1 - \xi)X_k']' - \lambda_k X_k = 0 \qquad (4)$$

and corresponding boundary conditions, for $\xi=0$:

$$X_k'' = 0, \quad X_k''' + \phi(1+\mu_1)X_k' + \left(\nu - \mu_0\lambda_k - \frac{\nu^2}{\nu - \mu_2\lambda_k}\right)X_k = 0, \quad L_k = \frac{\nu X_k}{\nu - m_2\lambda_k};$$

for $\xi = 1 \quad X_k'' = 0, \quad X_k''' + \phi\mu_1 X_k' + \mu_1\lambda_k X_k = 0,$ where

$$\xi = \frac{x}{l}, \quad \mu_0 = \frac{m_0}{m}, \quad \mu_1 = \frac{m_1}{m}, \quad \mu_2 = \frac{m_2}{m}, \quad \phi = \frac{mal^2}{EI}, \quad \lambda_k = \frac{w_k^2 ml^3}{EI}, \quad \nu = \frac{cl^3}{EI}, \quad \text{are}$$

dimensionless bow and arrow parameters, w_k are natural circular frequencies, λ_k are the eigenvalues.

Only in the simplest problem on static supported bars loaded by a constant axial force is a rigorous solution for lateral buckling known. To get approximate solutions we suggest an adoption of some shape for the arrow deflection curve that satisfies boundary conditions in (4). Solutions of equation (4) may be found in a form of the polynomial series like [1]:

$$X_k = \sum_{i=0}^{n} B_i \xi^i , \qquad (5)$$

where B_i are independent coefficients.

After substituting (5) in (4) and with intermediate transformations we get a linear algebraic system of recurrent equations with respect to L_k and B_1.

3.Natural Frequencies and Modes

Data results for the first four natural circular frequencies (in dimensionless form $\omega_k = \sqrt{\lambda}_k$) and modes have been obtained using 21 terms of the series (5) reducing the values of the last terms to computer zero when initial coefficients have been adapted with L_k=1. The results in graphical form for modern sport bow parameters are presented Fig.3, Fig.4.

Looking at the natural frequencies we notice that as the string force ϕ increases, the natural frequency ω decreases (see unbroken lines in the Fig.3). The lowest (named 'zero') frequency represents a turning motion of the arrow. Its eigenvalue increases from zero to negative values as the string force increases. Therefore, this frequency has zero or imaginary values, which means a monotone decrease of the arrow turn. The second natural frequency has only real values. It represents periodical oscillations of the bow and arrow system mainly as a system of rigid bodies. All other natural frequencies have both real and imaginary values because their eigenvalues have positive and negative values. The first (named 'main') circular frequency becomes zero as the string force ϕ=14-15. At this point the arrow loses its stability, because periodical oscillation transforms to a monotone increase of the arrow bend that deviates from the assigned direction. The third natural frequency reaches zero, and the point of elastic instability, at ϕ=56-57. The higher natural frequencies also decrease, as the force increases and have zero and imaginary values. They have not been plotted in Fig.3 because their influence in the whole bow and arrow lateral motion is negligibly small [5].

Conclusions

Theoretical investigation of the bow and arrow system can be separated into two parts. An arrow's lateral deflection is determined by the order of its longitudinal motion as independent of the deflection. The results of the data analysis of this problem show that errors that have been caused by a separation of the whole system into two parts do not exceed one percent of the whole energy of the system. A bow force increase causes a decrease in arrow natural frequencies. As the force reaches the critical value the frequency reduces to zero and the arrow loses its dynamic stability. Its periodical oscillations transform to a monotone increase of the arrow bend that deviates from the assigned direction. Imaginary values of natural frequencies do not depend on the bow parameters. The bow parameters have a significant influence on the real values of the arrow natural frequencies. For other equal conditions the bow and string stiffness influences the bend oscillations of the arrow more significantly than the bow mass. For proper bow-arrow matching, an arrow should have completed nearly one cycle of vibration in the time of its common motion with the string. The spine should be directly proportional to the mass of the arrow.

References

1. Feodos'ev V.I. (1965) About one problem on stability. *Applied mathematics and mechanics*, **29**, 391-392 (in Russian).
2. Hickman, C.N., Nagler, F. & Klopsteg, P.E. (1947) *Archery: the technical side.* National Field Archery Association, Radlands.
3. Lavrent'ev, M.A. & Ishlinsky, A.Y. (1949) The dynamic modes of buckling of elastic systems. *Papers of the Academy of Science of the USSR*, **64**, 779-782 (in Russian).
4. Shuster, B.G. (1969) Ballistics of the modern working recurve bow and arrow. *American Journal of Physics*, **37**, 364-373.
5. Zanevskyy, I.P. (1996) *The Methods of Simulation and Analysis of Sport Archery Parameters*. Dsc thesis, Physicomechanical Institute, Lviv (in Ukrainian, summery in English).

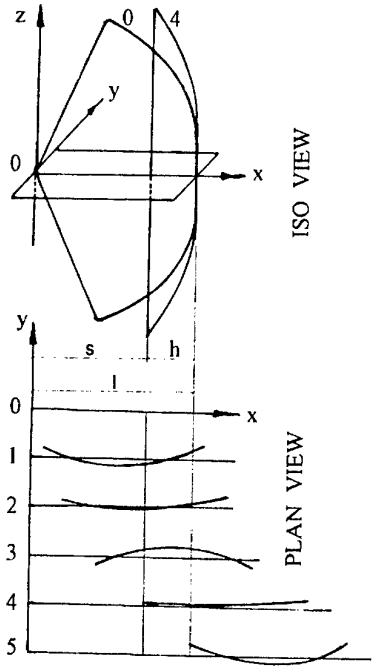


Figure 1 Diagram of the internal ballistics of the arrow: x-y-z is a rectangular co-ordinate system; x-z is a vertical plane of the bow string; 0, 1, 2, 3, 4, 5 are the successive positions of the arrow; 0 - is the initial position when archer releases string; 1, 2, and 3 are the positions during the common motion of the string and the arrow, 4 is the position when the arrow leaves the string ; 5 - is the position when the arrow leaves the bow area.

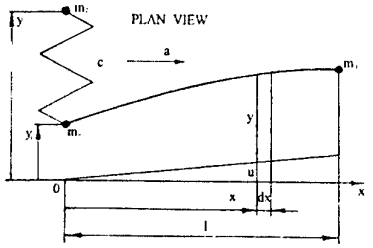


Figure 2 Schematic model of the arrow-bow system to investigate the lateral motion.

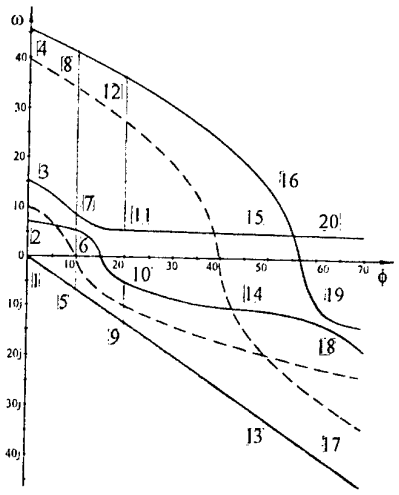


Figure 3 Four first natural circular frequencies of the arrow-bow system with the average sport target archery parameters (unbroken lines): μ_0=0.2, μ_1=0.25, μ_2=2.5, υ=20. Corresponding natural modes marked the same numbers have been presented in the Fig.4. The two first natural circular frequencies of the arrow under the buckling test are ploted with the dotted lines.

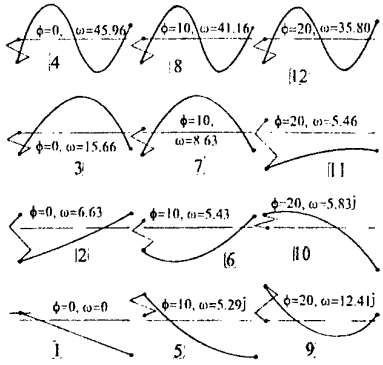


Figure 4 Natural modes of the arrow-bow system (see Fig.3) as the arrow leaves the string (ϕ=0), below the first elastic non-stability (ϕ=10) and over that (ϕ=20).